TUGAS PROYEK
APLIKASI KOMPUTER

Disusun oleh:
Wahyu Rananda Westri
22305144039

PROGRAM STUDI MATEMATIKA
JURUSAN PENDIDIKAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI YOGYAKARTA

2023

Daftar Isi

BAB 1

PERHITUNGAN ALJABAR DENGAN EMT

EMT untuk Perhitungan Aljabar

Pada notebook ini Anda belajar menggunakan EMT untuk melakukan berbagai perhitungan terkait dengan materi atau topik dalam Aljabar. Kegiatan yang harus Anda lakukan adalah sebagai berikut:

- Membaca secara cermat dan teliti notebook ini;
- Menerjemahkan teks bahasa Inggris ke bahasa Indonesia;
- Mencoba contoh-contoh perhitungan (perintah EMT) dengan cara meng-ENTER setiap perintah EMT yang ada (pindahkan kursor ke baris perintah)
- Jika perlu Anda dapat memodifikasi perintah yang ada dan memberikan keterangan/penjelasan tambahan terkait hasilnya.
- Menyisipkan baris-baris perintah baru untuk mengerjakan soal-soal Aljabar dari file PDF yang saya berikan;
- Memberi catatan hasilnya.
- Jika perlu tuliskan soalnya pada teks notebook (menggunakan format LaTeX).
- Gunakan tampilan hasil semua perhitungan yang eksak atau simbolik dengan format LaTeX. (Seperti contoh-contoh pada notebook ini.)

Contoh pertama

Menyederhanakan bentuk aljabar:

$$6x^{-3}y^5 \times -7x^2y^{-9}$$

```
>$&6*x^(-3)*y^5*-7*x^2*y^(-9)
```

$$-\frac{42}{x\,y^4}$$

Menjabarkan:

$$(6x^{-3}+y^5)(-7x^2-y^{-9})$$

```
>$&showev('expand((6*x^(-3)+y^5)*(-7*x^2-y^(-9))))
```

$$expand\left(\left(-\frac{1}{y^9}-7\,x^2\right)\,\left(y^5+\frac{6}{x^3}\right)\right)=-7\,x^2\,y^5-\frac{1}{y^4}-\frac{6}{x^3\,y^9}-\frac{42}{x}$$

Baris Perintah

Baris perintah Euler terdiri dari satu atau beberapa perintah Euler yang diikuti oleh titik koma ";" atau koma ",". Titik koma mencegah pencetakan hasil. Koma setelah perintah terakhir dapat dihilangkan.

Baris perintah berikut hanya akan mencetak hasil ekspresi, bukan tugas atau perintah.

```
>r:=2; h:=4; pi*r^2*h/3
```

```
16.7551608191
```

Perintah harus dipisahkan dengan spasi. Baris perintah berikut mencetak dua hasilnya.

```
>pi*2*r*h, %+2*pi*r*h // Ingat tanda % menyatakan hasil perhitungan terakhi
```

```
50.2654824574
100.530964915
```

Garis perintah dilaksanakan dalam urutan pengembalian pengguna. Jadi anda mendapatkan nilai baru setiap kali anda menjalankan baris kedua.

```
>x := 1;
>x := cos(x) // nilai cosinus (x dalam radian)
```

```
0.540302305868
```

```
>x := cos(x)//hasil dari cos(cos(1))
```

```
0.857553215846
```

Jika dua baris terhubung dengan "..." maka kedua baris tersebut akan selalu dieksekusi secara bersamaan.

Contoh lain:

```
>y=12; ...
>(y^2+6)/3,
```

```
50
```

```
>a=1; b=2; c=3; ...
>((24*(a^10)*(b^(-8))*(c^7))/12*(a^6)*(b^(-3))*c^5)^(-5)
```

```
0
```

```
>x := 1.5; ...
>x := (x+2/x)/2, x := (x+2/x)/2, x := (x+2/x)/2,
```

```
1.41666666667
1.41421568627
1.41421356237
```

Ini juga merupakan cara yang baik untuk membagi perintah panjang ke dalam dua baris atau lebih. Anda dapat menekan Ctrl+Return untuk membagi satu baris menjadi dua di posisi kursor saat ini, atau Ctrl+Back untuk menggabungkan baris-baris tersebut.

Untuk melipat semua baris multi, tekan Ctrl+L. Kemudian baris-baris berikutnya hanya akan terlihat jika salah satunya memiliki fokus. Untuk melipat satu baris multi, mulailah baris pertama dengan "%+".

```
>%+ x=4+5; ...
```

Sebuah baris yang dimulai dengan %% akan sepenuhnya tidak terlihat.

```
81
```

Contoh lain

```
0
```

Euler mendukung perulangan dalam baris perintah, selama perulangan tersebut cukup untuk satu baris atau beberapa baris. Di dalam program, pembatasan ini tidak berlaku, tentu saja. Untuk informasi lebih lanjut, konsultasikan pengenalan berikut ini.

```
>x=1; for i=1 to 5; x := (x+2/x)/2, end; // menghitung akar 2
```

```
1.5
1.41666666667
1.41421568627
1.41421356237
1.41421356237
```

Boleh menggunakan beberapa baris. Pastikan baris berakhir dengan "...".

```
>x := 1.5; // comments go here before the ...
>repeat xnew:=(x+2/x)/2; until xnew~=x; ...
>   x := xnew; ...
>end; ...
>x,
```

```
1.41421356237
```

Struktur kondisional juga berfungsi.

```
>if E^pi>pi^E; then "Thought so!", endif;
```

```
Thought so!
```

Contoh lain

```
>if (((2^6)*(2^(-3)))/((2^10)/2^(-8)))<1; then "yes", endif;
```

```
yes
```

Ketika Anda menjalankan suatu perintah, kursor dapat berada pada posisi apa pun dalam baris perintah. Anda dapat kembali ke perintah sebelumnya atau melompat ke perintah berikutnya dengan tombol panah. Atau Anda dapat mengklik ke dalam bagian komentar di atas perintah untuk menuju ke perintah tersebut.

Ketika Anda memindahkan kursor sepanjang baris, pasangan tanda kurung buka dan tutup akan disorot. Juga, perhatikan baris status. Setelah tanda kurung buka dari fungsi sqrt(), baris status akan menampilkan teks bantuan untuk fungsi tersebut. Jalankan perintah dengan tombol enter.

```
>sqrt(sin(10°)/cos(20°))
```

```
0.429875017772
```

Untuk melihat bantuan untuk perintah terbaru, buka jendela bantuan dengan F1. Di sana, Anda dapat memasukkan teks untuk mencari informasi. Pada baris kosong, bantuan untuk jendela bantuan akan ditampilkan. Anda dapat menekan tombol escape untuk menghapus baris atau untuk menutup jendela bantuan.

Anda dapat mengklik dua kali pada setiap perintah untuk membuka bantuan untuk perintah tersebut. Cobalah mengklik dua kali perintah 'exp' di bawah ini dalam baris perintah.

```
>exp(log(2.5))
```

```
2.5
```

Anda juga dapat menyalin dan menempel di Euler. Gunakan Ctrl-C dan Ctrl-V untuk ini. Untuk menandai teks, seret mouse atau gunakan tombol shift bersama dengan tombol kursor mana pun. Selain itu, Anda dapat menyalin tanda kurung yang disorot.

Syntax Dasar

"Euler mengenal fungsi matematika yang umum digunakan. Seperti yang Anda lihat di atas, fungsi trigonometri berfungsi dalam radian atau derajat. Untuk mengkonversi ke derajat, tambahkan simbol derajat (dengan tombol F7) ke nilai tersebut, atau gunakan fungsi rad(x). Fungsi akar kuadrat disebut sqrt dalam Euler. Tentu saja, x^(1/2) juga mungkin.

Untuk mengatur variabel, gunakan baik "=" atau ":=". Untuk kejelasan, pengantar ini menggunakan bentuk terakhir. Spasi tidak masalah. Tetapi ada harapan adanya spasi antara perintah-perintah.

Beberapa perintah dalam satu baris dipisahkan dengan "," atau ";". Semicolon akan menghilangkan output dari perintah tersebut. Di akhir baris perintah, tanda koma "," diasumsikan jika tanda titik koma ";" tidak ada.

```
>g:=9.81; t:=2.5; 1/2*g*t^2
```

```
30.65625
```

Contoh lain

```
>x:=2; y:=5; (6*x*y^3)*(9*(x^4)*y^2),
```

```
5400000
```

EMT menggunakan sintaks pemrograman untuk ekspresi. Untuk memasukkan

$$e^2 \cdot \left(\frac{1}{3+4\log(0.6)} + \frac{1}{7}\right)$$

anda harus mengatur tanda kurung yang benar dan menggunakan / untuk pecahan. Perhatikan tanda kurung yang disorot untuk bantuan. Perhatikan bahwa konstanta Euler e dinamai E dalam EMT.

```
>E^2*(1/(3+4*log(0.6))+1/7)
```

```
8.77908249441
```

Contoh lain

```
>a=5; b=10; n=2; ((a*n)+(b^n))/((a^n)-(b^n)),
```

```
-1.46666666667
```

Untuk menghitung ekspresi yang rumit seperti

$$\left(\frac{\frac{1}{7} + \frac{1}{8} + 2}{\frac{1}{3} + \frac{1}{2}}\right)^2 \pi$$

anda perlu memasukkannya dalam bentuk baris.

```
>((1/7 + 1/8 + 2) / (1/3 + 1/2))^2 * pi
```

```
23.2671801626
```

Contoh lain

```
>((4*(8-6)^2 + 4)*(3-2*8))/((2^2)*(2^3+5))
```

```
-5
```

Letakkan tanda kurung dengan hati-hati di sekitar sub-ekspresi yang perlu dihitung terlebih dahulu. EMT akan membantu Anda dengan cara menyoroti ekspresi yang ditutup oleh tanda kurung penutup. Anda juga harus memasukkan nama "pi" untuk huruf Yunani pi.

Hasil dari perhitungan ini adalah bilangan pecahan. Secara default, bilangan ini dicetak dengan sekitar 12 digit akurasi. Dalam baris perintah berikutnya, kita juga akan mempelajari bagaimana kita dapat merujuk ke hasil sebelumnya dalam baris yang sama

```
>1/3+1/7, fraction %
```

```
0.47619047619
10/21
```

Contoh lain

```
>1/5+1/10-1/2, fraction %
```

```
-0.2
-1/5
```

Sebuah perintah Euler dapat berupa ekspresi atau perintah primitif. Sebuah ekspresi terdiri dari operator dan fungsi. Jika diperlukan, ekspresi harus mengandung tanda kurung untuk memaksakan urutan eksekusi yang benar. Dalam keraguan, menetapkan tanda kurung adalah ide yang baik. Perlu diingat bahwa EMT menampilkan tanda kurung pembuka dan penutup saat mengedit baris perintah.

```
>(cos(pi/4)+1)^3*(sin(pi/4)+1)^2
```

```
14.4978445072
```

Operator numerik dalam Euler termasuk

operator plus uner atau unary
operator minus uner atau unary
*, /
. produk matriks
a^b pangkat untuk a positif atau b integer (a**b juga berfungsi)
n! operator faktorial
dan banyak lagi.

Berikut beberapa fungsi yang mungkin Anda butuhkan. Ada banyak lagi.

sin, cos, tan, atan, asin, acos, rad, deg
log, exp(fungsi eksponensial), log10, sqrt, logbase
bin, logbin, logfac, mod, floor, ceil, round, abs, sign
conj, re, im, arg, conj, real, kompleks
beta, betai, gamma, kompleksgamma, ellrf, ellf, ellrd, elle
bitand, bitor, bitxor, bitnot

Beberapa perintah memiliki alias, misalnya ln untuk log.

```
>ln(E^2), arctan(tan(0.5))
```

```
2
0.5
```

```
>sin(30°)
```

```
0.5
```

Contoh lain

$$log_3(81*27)$$

```
>logbase(81*27,3)
```

```
7
```

Contoh lain

$$\ln 5/2$$

```
>ln(5/2)
```

```
0.916290731874
```

Pastikan untuk menggunakan tanda kurung (tanda kurung bulat) setiap kali ada keraguan tentang urutan eksekusi! Yang berikut ini tidak sama dengan (2^3)^4, yang adalah default untuk 2^3^4 dalam EMT (beberapa sistem numerik melakukannya sebaliknya).

```
>2^3^4, (2^3)^4, 2^(3^4)
```

```
2.41785163923e+24
4096
2.41785163923e+24
```

Bilangan Real

Tipe data utama dalam Euler adalah bilangan real. Bilangan real direpresentasikan dalam format IEEE dengan akurasi sekitar 16 digit desimal.

Catatan tambahan:
Bilangan real meliputi bilangan rasional, seperti bilangan bulat 42 dan pecahan -23/129, dan bilangan irasional, seperti

$$\sqrt{2} dan \pi.$$

Bilangan real juga dapat dilambangkan sebagai salah satu titik dalam garis bilangan.

```
>longest 1/3
```

```
0.3333333333333333
```

Contoh lain

```
>longest 13/17
```

```
0.7647058823529411
```

```
>shortest 13/17
```

```
0.76
```

```
>longest 10/3
```

```
3.333333333333333
```

Representasi ganda internal menggunakan 8 byte.

```
>printdual(1/3)//Mencetak bilangan real x dengan mantisa ganda.
```

```
1.0101010101010101010101010101010101010101010101010101*2^-2
```

```
>printhex(1/3)//Mencetak bilangan real x dengan mantisa heksadesimal.
```

```
5.5555555555554*16^-1
```

Contoh lain

```
>printdual(13/17)
```

```
1.1000011110000111100001111000011110000111100001111000*2^-1
```

```
>printhex(10/3)
```

```
3.5555555555556*16^0
```

String

Dalam Euler, string didefinisikan dengan "..."

```
>"A string can contain anything."
```

```
A string can contain anything.
```

Contoh lain

```
>"Nama saya Wahyu Rananda Westri."
```

```
Nama saya Wahyu Rananda Westri.
```

String dapat digabungkan dengan | atau dengan +. Ini juga berlaku untuk angka, yang dikonversi menjadi string dalam kasus tersebut.

```
>"The area of the circle with radius " + 2 + " cm is " + pi*4 + " cm^2."
```

```
The area of the circle with radius 2 cm is 12.5663706144 cm^2.
```

Contoh lain

```
>"Nama saya Wahyu Rananda Westri. " + "Saya sekarang berumur" +" " + 20 + "
```

```
Nama saya Wahyu Rananda Westri. Saya sekarang berumur 20 tahun.
```

Fungsi print juga mengkonversi angka menjadi string. Ini dapat mengambil sejumlah digit dan sejumlah tempat (0 untuk output yang padat), dan jika memungkinkan satuan.

```
>"Golden Ratio : " + print((1+sqrt(5))/2,5,0)
```

```
Golden Ratio : 1.61803
```

Contoh lain

```
>"Rata-rata tinggi badan mahasiswa adalah"+ print((156+170+180+160+165)/5)
```

```
Rata-rata tinggi badan mahasiswa adalah    166.20
```

Ada sebuah string khusus yang disebut 'none', yang tidak dicetak. Ini dikembalikan oleh beberapa fungsi ketika hasilnya tidak penting. (Ini dikembalikan secara otomatis jika fungsi tersebut tidak memiliki pernyataan pengembalian.)

```
>none
```

Untuk mengkonversi sebuah string menjadi angka, cukup evaluasi string tersebut. Ini juga berlaku untuk ekspresi (lihat di bawah).

```
>"1234.5"()
```

```
1234.5
```

Untuk mendefinisikan vektor string, gunakan notasi vektor [...].

```
>v:=["affe","charlie","bravo"]
```

```
affe
charlie
bravo
```

Contoh lain

```
>ternak:=["ayam","kambing","sapi","bebek"]
```

```
ayam
kambing
sapi
bebek
```

Vektor string kosong ditunjukkan dengan [none]. Vektor string dapat digabungkan.

```
>w:=[none]; w|v|v
```

```
affe
charlie
bravo
affe
charlie
bravo
```

Contoh lain

```
>z=[none]; z|ternak|v
```

```
ayam
kambing
sapi
bebek
affe
charlie
bravo
```

String dapat mengandung karakter Unicode. Secara internal, string-string ini mengandung kode UTF-8. Untuk menghasilkan string semacam itu, gunakan u"..." dan salah satu entitas HTML.

String Unicode dapat digabungkan seperti string lainnya.

```
>u"&alpha; = " + 45 + u"&deg;" // pdfLaTeX mungkin gagal menampilkan secara
```

```
 = 45°
```

I

Dalam komentar, entitas yang sama seperti a, ß dll. dapat digunakan. Ini mungkin merupakan alternatif cepat untuk Latex. (Lebih banyak detail tentang komentar di bawah).

Ada beberapa fungsi untuk membuat atau menganalisis string Unicode. Fungsi strtochar() akan mengenali string Unicode dan menerjemahkannya dengan benar.

```
>v=strtochar(u"&Auml; is a German letter")//strtochar adalah fungsi untuk m
```

```
[196,  32,  105,  115,  32,  97,  32,  71,  101,  114,  109,  97,  110,
32,  108,  101,  116,  116,  101,  114]
```

Hasilnya adalah vektor angka Unicode. Fungsi kebalikannya adalah chartoutf().

```
>v[1]=strtochar(u"&Uuml;")[1]; chartoutf(v)
```

```
Ü is a German letter
```

Fungsi utf() dapat menerjemahkan sebuah string dengan entitas menjadi string Unicode dalam sebuah variabel.

```
>s="We have &alpha;=&beta;."; utf(s) // pdfLaTeX mungkin gagal menampilkan
```

```
We have =.
```

Juga memungkinkan untuk menggunakan entitas numerik.

```
>u"&#196;hnliches"//maksud u"&#196; merujuk pada karakter "Ä" (A dengan uml
```

```
Ähnliches
```

Nilai Boolean

Nilai Boolean direpresentasikan dengan 1=true atau 0=false dalam Euler. String dapat dibandingkan, sama seperti angka

```
>2<1, "apel"<"banana"
```

```
0
1
```

Catatan tambahan :
"apel"<"banana" karena dalam urutan leksikografi, "apel" berada sebelum "banana" karena "a" lebih awal dalam alfabet daripada "b". Jadi, perbandingan ini menghasilkan nilai True, yang menunjukkan bahwa "apel" kurang dari "banana" dalam urutan leksikografi.
Contoh lain:

```
>1/7>2/19
```

```
1
```

```
>"ayam">"jerapah"
```

```
0
```

Operator 'and' adalah '&&' dan 'or' adalah '||', seperti dalam bahasa C. (Kata-kata 'and' dan 'or' hanya dapat digunakan dalam kondisi 'if'.)

```
>2<E && E<3//Nilai "E" sekitar 2.71828
```

```
1
```

Contoh lain :

```
>"ayam"<"jerapah" && 1/7<2/19
```

```
0
```

```
>"ayam"<"jerapah" || 1/7<2/19
```

```
1
```

Operator boolean mengikuti aturan bahasa matriks.

```
>(1:10)>5, nonzeros(%)//nonzeroes(%) menghasilkan daftar yang berisi semua
```

```
[0,  0,  0,  0,  0,  1,  1,  1,  1,  1]
[6,  7,  8,  9,  10]
```

Anda dapat menggunakan fungsi nonzeros() untuk mengekstrak elemen-elemen tertentu dari sebuah vektor. Dalam contoh ini, kami menggunakan kondisional isprime(n).

```
>N=2|3:2:99 // N berisi elemen 2 dan bilangan2 ganjil dari 3 s.d. 99
```

```
[2,  3,  5,  7,  9,  11,  13,  15,  17,  19,  21,  23,  25,  27,  29,
31,  33,  35,  37,  39,  41,  43,  45,  47,  49,  51,  53,  55,  57,
59,  61,  63,  65,  67,  69,  71,  73,  75,  77,  79,  81,  83,  85,
87,  89,  91,  93,  95,  97,  99]
```

```
>N[nonzeros(isprime(N))] //pilih anggota2 N yang prima
```

```
[2,  3,  5,  7,  11,  13,  17,  19,  23,  29,  31,  37,  41,  43,  47,
53,  59,  61,  67,  71,  73,  79,  83,  89,  97]
```

Contoh lain

```
>M=3:3:100
```

```
[3,  6,  9,  12,  15,  18,  21,  24,  27,  30,  33,  36,  39,  42,  45,
48,  51,  54,  57,  60,  63,  66,  69,  72,  75,  78,  81,  84,  87,
90,  93,  96,  99]
```

Output Formats

Format keluaran default dari EMT mencetak 12 digit. Untuk memastikan bahwa kita melihat format default, kita mengatur ulang formatnya.

```
>defformat; pi
```

```
3.14159265359
```

Secara internal, EMT menggunakan standar IEEE untuk angka ganda dengan sekitar 16 digit desimal. Untuk melihat jumlah digit yang penuh, gunakan perintah "longestformat", atau gunakan operator "longest" untuk menampilkan hasil dalam format terpanjang.

```
>longest pi
```

```
      3.141592653589793
```

Contoh lain

```
>defformat; 127/17
```

```
7.47058823529
```

```
>longest 127/17
```

```
      7.470588235294118
```

Berikut adalah representasi heksadesimal internal dari angka ganda.

```
>printhex(pi)
```

```
3.243F6A8885A30*16^0
```

Contoh lain

```
>printhex(127/17)
```

```
7.7878787878788*16^0
```

Format keluaran dapat diubah secara permanen dengan perintah format.

```
>format(12,5); 1/3, pi, sin(1)//artinya totalada 12 digit angka dan 5 diant
```

```
    0.33333
    3.14159
    0.84147
```

Contoh lain

```
>format(12,5); 123456789/17
```

```
7262164.05882
```

Format default adalah format(12).

```
>format(12); 1/3
```

```
0.333333333333
```

Fungsi-fungsi seperti "shortestformat", "shortformat", "longformat" bekerja untuk vektor dengan cara berikut.

```
>shortestformat; random(3,8)
```

```
  0.66    0.2   0.89   0.28   0.53   0.31   0.44    0.3
  0.28   0.88   0.27    0.7   0.22   0.45   0.31   0.91
  0.19   0.46  0.095    0.6   0.43   0.73   0.47   0.32
```

Format default untuk skalar adalah format(12). Namun ini dapat diubah.

```
>setscalarformat(5); pi
```

```
3.1416
```

Contoh lain

```
>setscalarformat(3); 10/7
```

```
1.43
```

Fungsi "longestformat" juga mengatur format skalar.

```
>longestformat; pi
```

```
3.141592653589793
```

Untuk referensi, berikut adalah daftar format keluaran yang paling penting.

shortestformat
shortformat
longformat
longestformat
format(length,digits)
goodformat(length)
fracformat(length)
defformat
Akurasi internal EMT adalah sekitar 16 tempat desimal, sesuai dengan standar IEEE. Angka-angka disimpan dalam format internal ini.

Namun, format keluaran EMT dapat diatur secara fleksibel.

```
>longestformat; pi,
```

```
3.141592653589793
```

```
>format(10,5); pi
```

```
3.14159
```

Format default adalah defformat().

```
>defformat; // default
```

Ada operator-operator singkat yang hanya mencetak satu nilai. Operator "longest" akan mencetak semua digit valid dari sebuah angka.

```
>longest pi^2/2
```

```
4.934802200544679
```

Juga ada operator singkat untuk mencetak hasil dalam format pecahan. Kami telah menggunakannya di atas.

```
>fraction 1+1/2+1/3+1/4
```

```
25/12
```

Karena format internal menggunakan cara biner untuk menyimpan angka, nilai 0.1 tidak akan direpresentasikan secara tepat. Kesalahan tersebut akan terakumulasi sedikit, seperti yang Anda lihat dalam perhitungan berikut.

```
>longest 0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1-1
```

```
  -1.110223024625157e-16
```

Namun, dengan "longformat" default, Anda tidak akan melihat hal ini. Untuk kenyamanan, hasil keluaran dari angka yang sangat kecil adalah 0.

```
>0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1-1
```

```
 0
```

Expressions

String atau nama dapat digunakan untuk menyimpan ekspresi matematika, yang dapat dievaluasi oleh EMT. Untuk ini, gunakan tanda kurung setelah ekspresi. Jika Anda bermaksud menggunakan string sebagai ekspresi, gunakan konvensi untuk menamainya "fx" atau "fxy" dll. Ekspresi memiliki prioritas lebih tinggi dibandingkan fungsi.

Variabel global dapat digunakan dalam evaluasi.

```
>r:=2; fx:="pi*r^2"; longest fx()
```

```
       12.56637061435917
```

Contoh lain

```
>b=3; fx ="b^2 + E"; longest fx()
```

```
       11.71828182845904
```

Parameter diberikan kepada x, y, dan z sesuai urutan tersebut. Parameter tambahan dapat ditambahkan menggunakan parameter-parameter yang telah diberikan sebelumnya.

```
>fx:="a*sin(x)^2"; fx(5,a=-1)
```

```
 -0.919535764538
```

Contoh lain

```
>fx="c^2-cos(x)^2"; fx(3,b=1)
```

```
8.01991485667
```

Perhatikan bahwa ekspresi akan selalu menggunakan variabel global, bahkan jika ada variabel dalam fungsi dengan nama yang sama. (Sebaliknya, evaluasi ekspresi dalam fungsi dapat menghasilkan hasil yang sangat membingungkan bagi pengguna yang memanggil fungsi tersebut.)

```
>at:=4; function f(expr,x,at) := expr(x); ...
>f("at*x^2",3,5) // computes 4*3^2 not 5*3^2
```

```
36
```

Jika Anda ingin menggunakan nilai lain untuk "at" daripada nilai global, Anda perlu menambahkan "at=nilai".

```
>at:=4; function f(expr,x,a) := expr(x,at=a); ...
>f("at*x^2",3,5)
```

```
45
```

Sebagai referensi, kami mencatat bahwa koleksi panggilan (dibahas di tempat lain) dapat berisi ekspresi. Jadi, kita dapat membuat contoh di atas seperti berikut.

```
>at:=4; function f(expr,x) := expr(x); ...
>f({{"at*x^2",at=5}},3)
```

```
45
```

Ekspresi dalam x sering digunakan seperti fungsi.
Perlu diperhatikan bahwa mendefinisikan sebuah fungsi dengan nama yang sama seperti ekspresi simbolik global akan menghapus variabel ini untuk menghindari kebingungan antara ekspresi simbolik dan fungsi.

```
>f &= 5*x;
>function f(x) := 6*x;
>f(2)
```

```
12
```

Contoh lain

```
>f &=2*x;
>function f(x) := 3*x;
>f(-1)
```

```
-3
```

Sebagai konvensi, ekspresi simbolik atau numerik sebaiknya diberi nama fx, fxy, dll. Skema penamaan ini sebaiknya tidak digunakan untuk fungsi.

```
>fx &= diff(x^x,x); $&fx//diff digunakan untuk menghitung turunan
```

$$x^x\ (\log x + 1)$$

Contoh lain

```
>fx &= diff(x^3+2,x); $&fx
```

$$3\,x^2$$

Sebuah bentuk khusus dari ekspresi memungkinkan penggunaan variabel apa pun sebagai parameter tanpa nama untuk evaluasi ekspresi, bukan hanya "x", "y", dll. Untuk ini, mulailah ekspresi dengan "@(variabel) ...".

```
>"@(a,b) a^2+b^2", %(4,5)
```

```
@(a,b) a^2+b^2
41
```

Contoh lain

```
>"@(a,b) a^b + b^a", %(3,4)
```

```
@(a,b) a^b + b^a
145
```

Ini memungkinkan untuk memanipulasi ekspresi dalam variabel lain untuk fungsi-fungsi EMT yang membutuhkan ekspresi dalam "x".

Cara paling dasar untuk mendefinisikan fungsi sederhana adalah dengan menyimpan rumusnya dalam ekspresi simbolik atau numerik. Jika variabel utamanya adalah x, ekspresi tersebut dapat dievaluasi seperti sebuah fungsi.

Seperti yang Anda lihat dalam contoh berikut, variabel global terlihat selama evaluasi.

```
>fx &= x^3-a*x;  ...
>a=1.2; fx(0.5)
```

```
-0.475
```

Semua variabel lain dalam ekspresi dapat ditentukan dalam evaluasi menggunakan parameter yang telah ditentukan sebelumnya.

```
>fx(0.5,a=1.1)
```

```
-0.425
```

Sebuah ekspresi tidak perlu bersifat simbolik. Ini diperlukan jika ekspresi tersebut mengandung fungsi-fungsi yang hanya dikenal dalam kernel numerik, bukan dalam Maxima.

Matematika Simbolik

Matematika simbolik dalam EMT dilakukan dengan bantuan Maxima. Untuk detailnya, mulai dengan tutorial berikut ini, atau telusuri referensi Maxima. Para ahli dalam Maxima harus mencatat bahwa ada perbedaan dalam sintaksis antara sintaksis asli Maxima dan sintaksis default dalam ekspresi simbolik di EMT.

Matematika simbolik terintegrasi dengan lancar ke dalam Euler dengan tanda &. Setiap ekspresi yang dimulai dengan & adalah ekspresi simbolik. Itu dievaluasi dan dicetak oleh Maxima.

Pertama-tama, Maxima memiliki aritmatika "tak terbatas" yang dapat menangani angka-angka yang sangat besar.

```
>$&44!
```

$$2658271574788448768043625811014615890319638528000000000$$

Dengan cara ini, Anda dapat menghitung hasil yang besar secara tepat. Mari kita hitung

$$C(44,10) = \frac{44!}{34! \cdot 10!}$$

```
>$& 44!/(34!*10!) // nilai C(44,10)
```

$$2481256778$$

Contoh lain
Mari kita hitung

$$C(57,24) = \frac{57!}{33! \cdot 24!}$$

```
>$& 57!/(33!*24!)
```

$$7522327487513475$$

Tentu saja, Maxima memiliki fungsi yang lebih efisien untuk ini (seperti juga bagian numerik dari EMT).

```
>$binomial(44,10) //menghitung C(44,10) menggunakan fungsi binomial()
```

$$2481256778$$

Contoh lain

```
>$binomial(57,24)
```

$$7522327487513475$$

Untuk mempelajari lebih lanjut tentang fungsi tertentu, klik ganda pada fungsi tersebut. Misalnya, cobalah klik ganda pada "&binomial" dalam baris perintah sebelumnya. Ini akan membuka dokumentasi Maxima yang disediakan oleh para pengembang program tersebut.

Anda akan mengetahui bahwa yang berikut ini juga berfungsi.

$$C(x,3)=\frac{x!}{(x-3)!3!}=\frac{(x-2)(x-1)x}{6}$$

```
>$binomial(x,3) // C(x,3)
```

$$\frac{(x-2)\ (x-1)\ x}{6}$$

Contoh lain
Kita akan menghitung

$$C(a,4)=\frac{a!}{(a-3)!\cdot 3!}=\frac{a(a-1)(a-2)(a-3)(a-4)!}{(a-4)!\cdot 24}=\frac{a(a-1)(a-2)(a-3)}{24}$$

```
>$binomial(a,4)
```

$$\frac{(a-3)\ (a-2)\ (a-1)\ a}{24}$$

Jika Anda ingin menggantikan x dengan nilai tertentu, gunakan "with".

```
>$&binomial(x,3) with x=10 // substitusi x=10 ke C(x,3)
```

$$120$$

Contoh lain

```
>$&binomial(a,5) with a=20
```

$$15504$$

Dengan cara ini, Anda dapat menggunakan solusi dari suatu persamaan dalam persamaan lainnya.

Ekspresi simbolik dicetak oleh Maxima dalam bentuk 2D. Alasannya adalah ada tanda simbolik khusus dalam string tersebut.

Seperti yang Anda lihat dalam contoh-contoh sebelumnya dan berikutnya, jika Anda memiliki LaTeX terinstal, Anda dapat mencetak ekspresi simbolik dengan LaTeX. Jika tidak, perintah berikut akan menghasilkan pesan kesalahan.

Untuk mencetak ekspresi simbolik dengan LaTeX, gunakan $ di depan & (atau Anda dapat menghilangkan &) sebelum perintah. Jangan jalankan perintah Maxima dengan $ jika Anda tidak memiliki LaTeX terinstal.

```
>$(3+x)/(x^2+1)
```

$$\frac{x+3}{x^2+1}$$

Contoh lain

```
>$ (7-sqrt(-16))+(2+sqrt(-9))
```

$$9-i$$

```
>$ ((4-2*k)/(1+k))+((2-5*k)/(1+k))
```

$$\frac{4-2\,k}{k+1}+\frac{2-5\,k}{k+1}$$

Ekspresi simbolik dianalisis oleh Euler. Jika Anda membutuhkan sintaks yang kompleks dalam satu ekspresi, Anda dapat melampirkan ekspresi tersebut dalam "...". Menggunakan lebih dari satu ekspresi sederhana memungkinkan, tetapi sangat tidak disarankan.

```
>&"v := 5; v^2"
```

Untuk kelengkapan, kami mencatat bahwa ekspresi simbolik dapat digunakan dalam program, tetapi perlu diapit dengan tanda kutip. Selain itu, lebih efektif untuk memanggil Maxima pada saat kompilasi jika memungkinkan.

```
>$&expand((1+x)^4), $&factor(diff(%,x)) // diff: turunan, factor: faktor
```

$$x^4+4\,x^3+6\,x^2+4\,x+1$$

$$4\,\left(x+1\right)^3$$

Sekali lagi, % merujuk pada hasil sebelumnya.

Untuk memudahkan, kita simpan solusi ke dalam variabel simbolik. Variabel simbolik didefinisikan dengan "&=".

```
>fx &= (x+1)/(x^4+1); $&fx
```

$$\frac{x+1}{x^4+1}$$

Ekspresi simbolik dapat digunakan dalam ekspresi simbolik lainnya.

```
>$&factor(diff(fx,x))
```

$$\frac{-3\,x^4-4\,x^3+1}{\left(x^4+1\right)^2}$$

Input langsung perintah Maxima juga tersedia. Mulai baris perintah dengan "::". Sintaks Maxima disesuaikan dengan sintaks EMT (disebut "mode kompatibilitas").

```
>&factor(20!)
```

```
                    2432902008176640000
```

```
>:: factor(10!)
```

```
     8   4   2
    2   3   5   7
```

```
>:: factor(20!)
```

```
     18   8   4   2
    2    3   5   7   11  13  17  19
```

Contoh lain

```
>:: factor(12)
```

```
     2
    2   3
```

Jika Anda adalah ahli dalam Maxima, Anda mungkin ingin menggunakan sintaks asli Maxima. Anda dapat melakukannya dengan ":::".

```
>:: ::: av:g$ av^2;
>fx &= x^3*exp(x), $fx
```

```
     3   x
    x   E
```

$$x^3\,e^x$$

Variabel-variabel semacam itu dapat digunakan dalam ekspresi simbolik lainnya. Perhatikan bahwa dalam perintah berikut, sisi kanan dari &= dievaluasi sebelum penugasan ke Fx.

```
>&(fx with x=5), $%, &float(%)
```

```
        5
    125 E
```

$$125\,e^5$$

```
18551.64488782208
```

```
>fx(5)
```

```
18551.6448878
```

Untuk evaluasi suatu ekspresi dengan nilai-nilai tertentu dari variabel, Anda dapat menggunakan operator "with".

Baris perintah berikut juga menunjukkan bahwa Maxima dapat mengevaluasi suatu ekspresi secara numerik dengan float().

```
>&(fx with x=10)-(fx with x=5), &float(%)
```

```
      10         5
1000 E   - 125 E
```

```
2.20079141499189e+7
```

```
>$factor(diff(fx,x,2))
```

$$x\,\left(x^2+6\,x+6\right)\,e^{x}$$

Untuk mendapatkan kode LaTeX untuk suatu ekspresi, Anda dapat menggunakan perintah tex.

```
>tex(fx)
```

```
x^3\,e^{x}
```

Ekspresi simbolik dapat dievaluasi seperti ekspresi numerik.

```
>fx(0.5)
```

```
0.206090158838
```

Dalam ekspresi simbolik, ini tidak berfungsi, karena Maxima tidak mendukungnya. Sebaliknya, gunakan sintaks "with" (sebuah bentuk yang lebih baik dari perintah at(...) Maxima).

```
>$&fx with x=1/2
```

$$\frac{\sqrt{e}}{8}$$

Penugasan juga dapat bersifat simbolik.

```
>$&fx with x=1+t
```

$$(t+1)^3\ e^{t+1}$$

Perintah solve memecahkan ekspresi simbolik untuk sebuah variabel di Maxima. Hasilnya adalah vektor solusi.

```
>$&solve(x^2+x=4,x)
```

$$\left[x=\frac{-\sqrt{17}-1}{2}, x=\frac{\sqrt{17}-1}{2}\right]$$

Contoh lain

```
>$solve(x^2-2*x=15,x)
```

$$[x=-3, x=5]$$

Bandingkan dengan perintah "solve" numerik di Euler, yang memerlukan nilai awal dan opsionalnya sebuah nilai target.

```
>solve("x^2+x",1,y=4)
```

```
1.56155281281
```

Contoh lain

```
>solve("5*x^2+3*x",2,y=15)
```

```
1.45783958312
```

Nilai-nilai numerik dari solusi simbolik dapat dihitung dengan mengevaluasi hasil simbolik tersebut. Euler akan membaca penugasan x= dll. Jika Anda tidak memerlukan hasil numerik untuk perhitungan lebih lanjut, Anda juga dapat membiarkan Maxima menemukan nilai-nilai numeriknya.

```
>sol &= solve(x^2+2*x=4,x); $&sol, sol(), $&float(sol)
```

$$\left[x=-\sqrt{5}-1,x=\sqrt{5}-1\right]$$

```
[-3.23607,  1.23607]
```

$$[x=-3.23606797749979,x=1.23606797749979]$$

Untuk mendapatkan solusi simbolik tertentu, Anda dapat menggunakan "with" dan sebuah indeks.

```
>$&solve(x^2+x=1,x), x2 &= x with %[2]; $&x2
```

$$\left[x=\frac{-\sqrt{5}-1}{2},x=\frac{\sqrt{5}-1}{2}\right]$$

$$\frac{\sqrt{5}-1}{2}$$

Untuk menyelesaikan sebuah sistem persamaan, gunakan vektor persamaan. Hasilnya adalah vektor solusi.

```
>sol &= solve([x+y=3,x^2+y^2=5],[x,y]); $&sol, $&x*y with sol[1]
```

$$[[x=2,y=1]\,,[x=1,y=2]]$$

$$2$$

Ekspresi simbolik dapat memiliki flag, yang menunjukkan perlakuan khusus dalam Maxima. Beberapa flag dapat digunakan sebagai perintah juga, sementara yang lain tidak dapat. Flag-flag ini ditempatkan setelah "|" (sebuah bentuk yang lebih baik dari "ev(...,flags)").

```
>$& diff((x^3-1)/(x+1),x) //turunan bentuk pecahan
```

$$\frac{3\,x^2}{x+1}-\frac{x^3-1}{\left(x+1\right)^2}$$

```
>$& diff((x^3-1)/(x+1),x) | ratsimp //menyederhanakan pecahan
```

$$\frac{2\,x^3+3\,x^2+1}{x^2+2\,x+1}$$

```
>$&factor(%)
```

$$\frac{2\,x^3+3\,x^2+1}{\left(x+1\right)^2}$$

Functions

Dalam EMT, fungsi adalah program yang didefinisikan dengan perintah "function". Ini bisa menjadi fungsi satu baris atau fungsi multi-baris. Fungsi satu baris dapat berupa fungsi numerik atau simbolik. Fungsi satu baris numerik didefinisikan dengan ":=".

```
>function f(x) := x*sqrt(x^2+1)
```

Untuk gambaran umum, kami menampilkan semua definisi yang mungkin untuk fungsi satu baris. Sebuah fungsi dapat dievaluasi seperti fungsi Euler bawaan lainnya.

```
>f(2)
```

```
4.472135955
```

Fungsi ini akan berfungsi untuk vektor juga, mengikuti bahasa matriks Euler, karena ekspresi yang digunakan dalam fungsi tersebut di-vektorisasi.

```
>f(0:0.1:1)
```

```
[0,  0.100499,  0.203961,  0.313209,  0.430813,  0.559017,  0.699714,
0.854459,  1.0245,  1.21083,  1.41421]
```

Fungsi dapat diplot. Alih-alih ekspresi, kita hanya perlu memberikan nama fungsi.

Berbeda dengan ekspresi simbolik atau numerik, nama fungsi harus diberikan dalam bentuk string.

```
>solve("f",1,y=1)
```

```
0.786151377757
```

Secara default, jika Anda perlu menggantikan fungsi bawaan, Anda harus menambahkan kata kunci "overwrite". Menggantikan fungsi bawaan ini berbahaya dan dapat menyebabkan masalah bagi fungsi lain yang bergantung pada mereka.

Anda masih dapat memanggil fungsi bawaan sebagai "_...", jika itu adalah fungsi inti Euler.

```
>function overwrite sin (x) := _sin(x°) // redine sine in degrees
>sin(45)
```

```
0.707106781187
```

Lebih baik kita menghapus penggantian definisi sin ini.

```
>forget sin; sin(pi/4)
```

```
0.707106781187
```

Contoh lain

```
>function f(x):= 2*x-(9*sqrt(x))+4
>f(4)
```

```
-6
```

```
>f(7)
```

```
-5.81176179958
```

```
>function f(x):=(2*x-3)^2-5*(2*x-3)+6
>f(2)
```

```
2
```

```
>function overwrite cos (x) := _cos(x°)
>cos(90)
```

```
0
```

```
>forget cos; cos(pi/2)
```

```
0
```

Parameter Default

Fungsi numerik dapat memiliki parameter default.

```
>function f(x,a=1) := a*x^2
```

Menghilangkan parameter ini akan menggunakan nilai default.

```
>f(4)
```

```
16
```

Mengaturnya akan mengganti nilai default.

```
>f(4,5)
```

```
80
```

Parameter yang ditetapkan juga akan menggantinya. Ini digunakan oleh banyak fungsi Euler seperti plot2d, plot3d.

```
>f(4,a=1)
```

```
16
```

Contoh lain

```
>function f(x, a=3) := x^(1/2)-a*x^(1/4)+2
>f(2)
```

```
-0.153407782635
```

```
>f(1,4)
```

```
-1
```

```
>f(4,a=1)
```

```
2.58578643763
```

Jika sebuah variabel bukan parameter, maka variabel tersebut harus bersifat global. Fungsi satu baris dapat melihat variabel global.

```
>function f(x) := a*x^2
>a=6; f(2)
```

```
24
```

Namun, parameter yang ditetapkan akan mengesampingkan nilai global.

Jika argumen tidak ada dalam daftar parameter yang telah ditentukan sebelumnya, itu harus dideklarasikan dengan ":="!

```
>f(2,a:=5)
```

```
20
```

Contoh lain

```
>function f(x) := x^4 + a*x^2-5
>a=1; f(2)
```

```
15
```

```
>f(2, a:= 10)
```

```
51
```

Fungsi simbolik didefinisikan dengan "&=". Mereka didefinisikan dalam Euler dan Maxima, dan berfungsi di kedua dunia tersebut. Ekspresi yang digunakan untuk mendefinisikan dijalankan melalui Maxima sebelum definisi.

```
>function g(x) &= x^3-x*exp(-x); $&g(x)
```

$$x^3 - x\,e^{-x}$$

Fungsi simbolik dapat digunakan dalam ekspresi simbolik.

```
>$&diff(g(x),x), $&% with x=4/3
```

$$x\,e^{-x} - e^{-x} + 3\,x^2$$

$$\frac{e^{-\frac{4}{3}}}{3} + \frac{16}{3}$$

Contoh lain

```
>function f(x) &= x^4-6*x^3+8*x^2+6*x-9; $&f(x)
```

$$x^4-6\,x^3+8\,x^2+6\,x-9$$

```
>$&diff(f(x),x), $&% with x=2
```

$$4\,x^3-18\,x^2+16\,x+6$$

$$-2$$

Mereka juga dapat digunakan dalam ekspresi numerik. Tentu saja, ini hanya akan berfungsi jika EMT dapat menginterpretasikan semua yang ada di dalam fungsi tersebut.

```
>g(5+g(1))
```

```
178.635099908
```

Mereka dapat digunakan untuk mendefinisikan fungsi atau ekspresi simbolik lainnya.

```
>function G(x) &= factor(integrate(g(x),x)); $&G(c) // integrate: menginteg
```

$$\frac{e^{-c}\,\left(c^4\,e^{c}+4\,c+4\right)}{4}$$

Contoh lain

```
>function F(x) &= factor(integrate(g(x),x)); $&G(c)
```

$$\frac{e^{-c}\,\left(c^4\,e^{c}+4\,c+4\right)}{4}$$

```
>solve(&g(x),0.5)
```

```
0.703467422498
```

Berikut juga berfungsi, karena Euler menggunakan ekspresi simbolis dalam fungsi g, jika tidak menemukan variabel simbolis g, dan jika ada fungsi simbolis g.

```
>solve(&g,0.5)
```

```
0.703467422498
```

```
>function P(x,n) &= (2*x-1)^n; $&P(x,n)
```

$$(2\,x-1)^n$$

```
>function Q(x,n) &= (x+2)^n; $&Q(x,n)
```

$$(x+2)^n$$

```
>$&P(x,4), $&expand(%)
```

$$(2\,x-1)^4$$

$$16\,x^4-32\,x^3+24\,x^2-8\,x+1$$

```
>P(3,4)
```

```
625
```

```
>$&P(x,4)+ Q(x,3), $&expand(%)
```

$$(2\,x-1)^4+(x+2)^3$$

$$16\,x^4-31\,x^3+30\,x^2+4\,x+9$$

```
>$&P(x,4)-Q(x,3), $&expand(%), $&factor(%)
```

$$(2\,x-1)^4-(x+2)^3$$

$$16\,x^4-33\,x^3+18\,x^2-20\,x-7$$

$$16\,x^4-33\,x^3+18\,x^2-20\,x-7$$

```
>$&P(x,4)*Q(x,3), $&expand(%), $&factor(%)
```

$$(x+2)^3\,(2\,x-1)^4$$

$$16\,x^7+64\,x^6+24\,x^5-120\,x^4-15\,x^3+102\,x^2-52\,x+8$$

$$(x+2)^3\,(2\,x-1)^4$$

```
>$&P(x,4)/Q(x,1), $&expand(%), $&factor(%)
```

$$\frac{(2\,x-1)^4}{x+2}$$

$$\frac{16\,x^4}{x+2}-\frac{32\,x^3}{x+2}+\frac{24\,x^2}{x+2}-\frac{8\,x}{x+2}+\frac{1}{x+2}$$

$$\frac{(2\,x-1)^4}{x+2}$$

```
>function f(x) &= x^3-x; $&f(x)
```

$$x^3-x$$

Dengan &=, fungsi tersebut bersifat simbolis, dan dapat digunakan dalam ekspresi simbolis lainnya.

```
>$&integrate(f(x),x)
```

$$\frac{x^4}{4} - \frac{x^2}{2}$$

Dengan :=, fungsi tersebut bersifat numerik. Contoh yang baik adalah integral definitif seperti

$$f(x) = \int_1^x t^t \, dt,$$

Jika kita mendefinisikan ulang fungsi dengan kata kunci "map," itu dapat digunakan untuk vektor x. Secara internal, fungsi tersebut dipanggil untuk semua nilai x sekali, dan hasilnya disimpan dalam sebuah vektor.

```
>function map f(x) := integrate("x^x",1,x)
>f(0:0.5:2)
```

```
[-0.783431,  -0.410816,  0,  0.676863,  2.05045]
```

Fungsi dapat memiliki nilai default untuk parameter-parameternya.

```
>function mylog (x,base=10) := ln(x)/ln(base);
```

Sekarang, fungsi dapat dipanggil dengan atau tanpa parameter "base".

```
>mylog(100), mylog(2^6.7,2)
```

```
2
6.7
```

Selain itu, memungkinkan untuk menggunakan parameter yang sudah diassign.

```
>mylog(E^2,base=E)
```

```
2
```

Seringkali, kita ingin menggunakan fungsi untuk vektor di satu tempat, dan untuk elemen-elemen individual di tempat lain. Hal ini memungkinkan dengan menggunakan parameter vektor.

```
>function f([a,b]) &= a^2+b^2-a*b+b; $&f(a,b), $&f(x,y)
```

$$b^2 - a\,b + b + a^2$$

$$y^2 - x\,y + y + x^2$$

Fungsi simbolis seperti itu dapat digunakan untuk variabel-variabel simbolis.
Namun, fungsi tersebut juga dapat digunakan untuk vektor numerik.

```
>v=[3,4]; f(v)
```

```
17
```

Ada juga fungsi yang murni simbolis, yang tidak dapat digunakan secara numerik.

```
>function lapl(expr,x,y) &&= diff(expr,x,2)+diff(expr,y,2)//turunan parsial
```

```
diff(expr, y, 2) + diff(expr, x, 2)
```

```
>$&realpart((x+I*y)^4), $&lapl(%,x,y)
```

$$y^4 - 6\,x^2\,y^2 + x^4$$

$$0$$

Tentu saja, mereka dapat digunakan dalam ekspresi simbolis atau dalam definisi fungsi simbolis.

```
>function f(x,y) &= factor(lapl((x+y^2)^5,x,y)); $&f(x,y)
```

$$10\,\left(y^2 + x\right)^3\,\left(9\,y^2 + x + 2\right)$$

Untuk merangkum:

- &= mendefinisikan fungsi simbolis,
- := mendefinisikan fungsi numerik,
- &&= mendefinisikan fungsi murni simbolis.

Contoh soal lain

```
>function A(x,n) &= (x*4-2)^(2*n); $&A(x,n)
```

$$\left(4\,x-2\right)^{2\,n}$$

```
>function B(x,n) &= (x-1)^n; $&B(x,n)
```

$$\left(x-1\right)^{n}$$

```
>$&A(x,4), $&expand(%)
```

$$\left(4\,x-2\right)^{8}$$

$$65536\,x^{8}-262144\,x^{7}+458752\,x^{6}-458752\,x^{5}+286720\,x^{4}-114688\,x^{3}+28672\,x^{2}-4096\,x+256$$

```
>P(2,4)
```

```
81
```

```
>$&A(x,2)+ B(x,1), $&expand(%)
```

$$\left(4\,x-2\right)^{4}+x-1$$

$$256\,x^{4}-512\,x^{3}+384\,x^{2}-127\,x+15$$

```
>$&A(x,2)-B(x,1), $&expand(%), $&factor(%)
```

$$(4\,x-2)^4-x+1$$

$$256\,x^4-512\,x^3+384\,x^2-129\,x+17$$

$$256\,x^4-512\,x^3+384\,x^2-129\,x+17$$

```
>$&A(x,2)*B(x,1), $&expand(%), $&factor(%)
```

$$(x-1)\,(4\,x-2)^4$$

$$256\,x^5-768\,x^4+896\,x^3-512\,x^2+144\,x-16$$

$$16\,(x-1)\,(2\,x-1)^4$$

```
>function f(x) &= x^2+2*x
```

```
                                2
                               x  + 2 x
```

```
>$&integrate(f(x),x)
```

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

```
>function map f(x) := integrate("x^x",1,x)
>f(0:0.5:2)
```

```
 [-0.783431,  -0.410816,  0,  0.676863,  2.05045]
```

```
>function mylog (x,base=10) := ln(x)/ln(base);
>mylog(100), mylog(2^6.7,2)
```

```
2
6.7
```

```
>v=[3,4]; f(v)
```

```
[13.7251,  113.336]
```

Menyelesaikan Ekspresi

Ekspresi dapat dipecahkan secara numerik dan simbolis.

Untuk menyelesaikan ekspresi sederhana dengan satu variabel, kita dapat menggunakan fungsi solve(). Ini memerlukan nilai awal untuk memulai pencarian. Secara internal, solve() menggunakan metode sekant.

```
>solve("x^2-2",1)
```

```
1.41421356237
```

Ini juga berlaku untuk ekspresi simbolis. Ambil fungsi berikut sebagai contoh.

```
>$&solve(x^2=2,x)
```

$$\left[x=-\sqrt{2} , x=\sqrt{2} \right]$$

```
>$&solve(x^2-2,x)
```

$$\left[x=-\sqrt{2} , x=\sqrt{2} \right]$$

```
>$&solve(a*x^2+b*x+c=0,x)
```

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

```
>$&solve([a*x+b*y=c,d*x+e*y=f],[x,y])
```

$$\left[\left[x=-\frac{c\,e}{b\,\left(d-2\right)-a\,e} , y=\frac{c\,\left(d-2\right)}{b\,\left(d-2\right)-a\,e} \right] \right]$$

Contoh lain

```
>solve("x^3-2",1)
```

```
1.25992104989
```

```
>$&solve(x^3=2,x)
```

$$\left[x=\frac{2^{\frac{1}{3}}\,\sqrt{3}\,i-2^{\frac{1}{3}}}{2} , x=\frac{-2^{\frac{1}{3}}\,\sqrt{3}\,i-2^{\frac{1}{3}}}{2} , x=2^{\frac{1}{3}}\right]$$

```
>$&solve(x^3-2,x)
```

$$\left[x=\frac{2^{\frac{1}{3}}\,\sqrt{3}\,i-2^{\frac{1}{3}}}{2} , x=\frac{-2^{\frac{1}{3}}\,\sqrt{3}\,i-2^{\frac{1}{3}}}{2} , x=2^{\frac{1}{3}}\right]$$

```
>$&solve(x-(12/x)-a=0,x)
```

$$\left[x=\frac{a-\sqrt{a^2+48}}{2} , x=\frac{\sqrt{a^2+48}+a}{2}\right]$$

```
>px &= 4*x^8+x^7-x^4-x; $&px
```

$$4\,x^8+x^7-x^4-x$$

Sekarang kita mencari titik di mana polinomialnya bernilai 2. Dalam solve(), nilai target default y=0 dapat diubah dengan variabel yang ditugaskan. Kita menggunakan y=2 dan memeriksa dengan mengevaluasi polinomial pada hasil sebelumnya.

```
>solve(px,1,y=2), px(%)
```

```
0.966715594851
2
```

Menyelesaikan ekspresi simbolis dalam bentuk simbolis akan mengembalikan daftar solusi. Kami menggunakan pemecah masalah simbolis solve() yang disediakan oleh Maxima.

```
>sol &= solve(x^2-x-1,x); $&sol
```

$$\left[x=\frac{1-\sqrt{5}}{2},x=\frac{\sqrt{5}+1}{2}\right]$$

Cara tercepat untuk mendapatkan nilai numerik adalah dengan mengevaluasi solusinya secara numerik seperti ekspresi biasa.

```
>longest sol()
```

```
      -0.6180339887498949        1.618033988749895
```

Untuk menggunakan solusi secara simbolis dalam ekspresi lain, cara yang paling mudah adalah dengan menggunakan "with".

```
>$&x^2 with sol[1], $&expand(x^2-x-1 with sol[2])
```

$$\frac{\left(\sqrt{5}-1\right)^2}{4}$$

$$0$$

Menyelesaikan sistem persamaan secara simbolis dapat dilakukan dengan menggunakan vektor-vektor persamaan dan pemecah masalah simbolis solve(). Hasilnya adalah daftar dari daftar-daftar persamaan.

```
>$&solve([x+y=2,x^3+2*y+x=4],[x,y])
```

$$\left[\left[x=-1,y=3\right],\left[x=1,y=1\right],\left[x=0,y=2\right]\right]$$

Fungsi f() dapat melihat variabel global. Namun, seringkali kita ingin menggunakan parameter lokal.

$$a^x-x^a=0.1$$

dengan a=3.

```
>function f(x,a) := x^a-a^x;
```

Salah satu cara untuk meneruskan parameter tambahan ke f() adalah dengan menggunakan sebuah daftar yang berisi nama fungsi dan parameter-parameternya (cara lainnya adalah menggunakan parameter semikolon).

```
>solve({{"f",3}},2,y=0.1)
```

```
2.54116291558
```

Ini juga berfungsi dengan ekspresi. Namun, dalam hal ini, harus digunakan elemen daftar yang diberi nama. (Lebih lanjut tentang daftar dapat ditemukan dalam tutorial tentang sintaksis EMT).

```
>solve({{"x^a-a^x",a=3}},2,y=0.1)
```

```
2.54116291558
```

Contoh lain

```
>qx &= 2*x^4+x^3-x^2-x; $&qx
```

$$2\,x^4+x^3-x^2-x$$

```
>solve(qx,2,y=2), qx(%)
```

```
1.10455409197
2
```

```
>$&solve([x+2*y=2,x^2+2*y+2*x=4],[x,y])
```

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

Menyelesaikan Pertidaksamaan

Untuk menyelesaikan pertidaksamaan, EMT tidak akan dapat melakukannya, melainkan dengan bantuan Maxima, artinya secara eksak (simbolik). Perintah Maxima yang digunakan adalah fourier_elim(), yang harus dipanggil dengan perintah "load(fourier_elim)" terlebih dahulu.

```
>&load(fourier_elim)
```

```
          C:/Program Files/Euler x64/maxima/share/maxima/5.35.1/share/f\
  ourier_elim/fourier_elim.lisp
```

```
>$&fourier_elim([x^2 - 1>0],[x]) // x^2-1 > 0
```

$$[1 < x] \vee [x < -1]$$

```
>$&fourier_elim([x^2 - 1<0],[x]) // x^2-1 < 0
```

$$[-1 < x, x < 1]$$

```
>$&fourier_elim([x^2 - 1 # 0],[x]) // x^-1 <> 0
```

$$[-1 < x, x < 1] \vee [1 < x] \vee [x < -1]$$

```
>$&fourier_elim([x # 6],[x])
```

$$[x < 6] \vee [6 < x]$$

```
>$&fourier_elim([x < 1, x > 1],[x]) // tidak memiliki penyelesaian
```

$$emptyset$$

```
>$&fourier_elim([minf < x, x < inf],[x]) // solusinya R
```

$$universalset$$

```
>$&fourier_elim([x^3 - 1 > 0],[x])
```

$$\left[1<x,x^2+x+1>0\right]\vee\left[x<1,-x^2-x-1>0\right]$$

```
>$&fourier_elim([cos(x) < 1/2],[x]) // ??? gagal
```

$$[1-2\cos x>0]$$

```
>$&fourier_elim([y-x < 5, x - y < 7, 10 < y],[x,y]) // sistem pertidaksamaa
```

$$[y-5<x,x<y+7,10<y]$$

```
>$&fourier_elim([y-x < 5, x - y < 7, 10 < y],[y,x])
```

$$[max\left(10,x-7\right)<y,y<x+5,5<x]$$

```
>$&fourier_elim((x + y < 5) and (x - y >8),[x,y])
```

$$\left[y+8<x,x<5-y,y<-\frac{3}{2}\right]$$

```
>$&fourier_elim(((x + y < 5) and x < 1) or  (x - y >8),[x,y])
```

$$[y+8<x]\vee[x<min\left(1,5-y\right)]$$

```
>&fourier_elim([max(x,y) > 6, x # 8, abs(y-1) > 12],[x,y])
```

```
          [6 < x, x < 8, y < - 11] or [8 < x, y < - 11]
    or [x < 8, 13 < y] or [x = y, 13 < y] or [8 < x, x < y, 13 < y]
    or [y < x, 13 < y]
```

```
>$&fourier_elim([(x+6)/(x-9) <= 6],[x])
```

$$[x = 12] \vee [12 < x] \vee [x < 9]$$

Contoh lain

```
>$&fourier_elim([x^3 -8>0],[x])
```

$$\left[2 < x, x^2 + 2\,x + 4 > 0\right] \vee \left[x < 2, -x^2 - 2\,x - 4 > 0\right]$$

```
>$&fourier_elim([x # 1],[x])
```

$$[x < 1] \vee [1 < x]$$

```
>$&fourier_elim([x < 4, x > 4],[x])//tidak punya penyelesaian
```

$$emptyset$$

```
>$&fourier_elim([2*y-3*x < 7, 2*x - y < 10, 12 < y],[x,y])
```

$$\left[\frac{2\,y}{3} - \frac{7}{3} < x, x < \frac{y}{2} + 5, 12 < y, y < 44\right]$$

```
>$&fourier_elim(((3*x + y < 6) and x < 3) or  (x - 3*y >10),[x,y])
```

$$[3\,y+10<x]\vee\left[x<min\left(3,2-\frac{y}{3}\right)\right]$$

```
>$&fourier_elim([x^3 - 1\2 # 0],[x])
```

$$\left[12-x^3>0\right]\vee\left[x^3-12>0\right]$$

Bahasa Matriks

Dokumentasi inti EMT berisi diskusi terperinci mengenai bahasa matriks Euler.

Vektor dan matriks dimasukkan dengan tanda kurung siku, elemen-elemennya dipisahkan oleh koma, dan barisnya dipisahkan oleh titik koma.

```
>A=[1,2;3,4]
```

```
                    1                   2
                    3                   4
```

Hasil perkalian matriks dilambangkan dengan titik (dot).

```
>b=[3;4]
```

```
                    3
                    4
```

```
>b' // transpose b
```

```
  [3,   4]
```

```
>inv(A) //inverse A
```

```
                   -2                   1
                  1.5                -0.5
```

```
>A.b //perkalian matriks
```

```
              11
              25
```

```
>A.inv(A)
```

```
               1              0
               0              1
```

Poin utama dari bahasa matriks adalah bahwa semua fungsi dan operator bekerja pada setiap elemen secara individu.

```
>A.A
```

```
               7             10
              15             22
```

```
>A^2 //perpangkatan elemen2 A
```

```
               1              4
               9             16
```

```
>A.A.A
```

```
              37             54
              81            118
```

```
>power(A,3) //perpangkatan matriks
```

```
              37             54
              81            118
```

```
>A/A //pembagian elemen-elemen matriks yang seletak
```

```
               1              1
               1              1
```

```
>A/b //pembagian elemen2 A oleh elemen2 b kolom demi kolom (karena b vektor
```

```
    0.333333      0.666667
        0.75             1
```

```
>A\b // hasilkali invers A dan b, A^(-1)b
```

```
          -2
         2.5
```

```
>inv(A).b
```

```
          -2
         2.5
```

```
>A\A   //A^(-1)A
```

```
           1             0
           0             1
```

```
>inv(A).A
```

```
           1             0
           0             1
```

```
>A*A //perkalin elemen-elemen matriks seletak
```

```
           1             4
           9            16
```

Ini bukanlah perkalian matriks, melainkan perkalian elemen demi elemen. Hal yang sama berlaku untuk vektor.

```
>b^2 // perpangkatan elemen-elemen matriks/vektor
```

```
           9
          16
```

Jika salah satu operand adalah vektor atau skalar, maka operand tersebut diperluas dengan cara yang alami.

```
>2*A
```

```
              2              4
              6              8
```

Contohnya, jika operandnya adalah vektor kolom, elemennya diterapkan pada semua baris A.

```
>[1,2]*A
```

```
              1              4
              3              8
```

Jika itu adalah vektor baris, maka vektor tersebut diterapkan pada semua kolom A.

```
>A*[2,3]
```

```
              2              6
              6             12
```

Anda dapat membayangkan perkalian ini seolah-olah vektor baris v telah digandakan untuk membentuk matriks dengan ukuran yang sama dengan A.

```
>dup([1,2],2) // dup: menduplikasi/menggandakan vektor [1,2] sebanyak 2 kal
```

```
              1              2
              1              2
```

```
>A*dup([1,2],2)
```

```
              1              4
              3              8
```

Contoh lain

```
>C=[2,4,6,8;3,6,9,12]
```

```
            2             4             6             8
            3             6             9            12
```

```
>D=[1,2;2,1]
```

```
            1             2
            2             1
```

```
>C'
```

```
            2             3
            4             6
            6             9
            8            12
```

```
>inv(D)
```

```
    -0.333333      0.666667
     0.666667     -0.333333
```

```
>E=[3,2;4,3]
```

```
            3             2
            4             3
```

```
>D.E
```

```
           11             8
           10             7
```

```
>D.inv(D)
```

```
            1             0
            0             1
```

```
>E.E
```

```
        17          12
        24          17
```

```
>C^2
```

```
         4          16          36          64
         9          36          81         144
```

```
>power(D,4)
```

```
        41          40
        40          41
```

```
>D/E
```

```
  0.333333           1
       0.5    0.333333
```

```
>E\D
```

```
        -1           4
         2          -5
```

```
>C*C
```

```
         4          16          36          64
         9          36          81         144
```

```
>2*C
```

```
         4           8          12          16
         6          12          18          24
```

```
>D*[2,3]
```

```
                  2                 6
                  4                 3
```

```
>E*dup([1,2],2)
```

```
                  3                 4
                  4                 6
```

Ini juga berlaku untuk dua vektor di mana satu adalah vektor baris dan yang lainnya adalah vektor kolom. Kita dapat menghitung i*j untuk i dan j dari 1 hingga 5. Triknya adalah dengan mengalikan 1:5 dengan transposenya. Bahasa matriks Euler secara otomatis menghasilkan tabel nilai.

```
>(1:5)*(1:5)' // hasilkali elemen-elemen vektor baris dan vektor kolom
```

```
                  1                 2                 3                 4                 5
                  2                 4                 6                 8                10
                  3                 6                 9                12                15
                  4                 8                12                16                20
                  5                10                15                20                25
```

Sekali lagi, ingatlah bahwa ini bukanlah perkalian matriks!

```
>(1:5).(1:5)' // hasilkali vektor baris dan vektor kolom
```

```
  55
```

```
>sum((1:5)*(1:5)) // sama hasilnya
```

```
  55
```

Bahkan operator seperti < atau == bekerja dengan cara yang sama.

```
>(1:10)<6 // menguji elemen-elemen yang kurang dari 6
```

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

Misalnya, kita dapat menghitung jumlah elemen yang memenuhi kondisi tertentu dengan fungsi sum().

```
>sum((1:10)<6) // banyak elemen yang kurang dari 6
```

```
5
```

Euler memiliki operator perbandingan, seperti "==", yang memeriksa kesetaraan.

Kita mendapatkan vektor berisi 0 dan 1, di mana 1 mengindikasikan nilai benar (true).

```
>t=(1:10)^2; t==25 //menguji elemen2 t yang sama dengan 25 (hanya ada 1)
```

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

Dari vektor seperti itu, "nonzeros" memilih elemen-elemen yang bukan nol.

Dalam kasus ini, kita mendapatkan indeks dari semua elemen yang lebih besar dari 50.

```
>nonzeros(t>50) //indeks elemen2 t yang lebih besar daripada 50
```

```
[8,  9,  10]
```

Tentu saja, kita dapat menggunakan vektor ini untuk mengambil nilai-nilai yang sesuai dalam t.

```
>t[nonzeros(t>50)] //elemen2 t yang lebih besar daripada 50
```

```
[64,  81,  100]
```

Sebagai contoh, mari temukan semua kuadrat dari angka 1 hingga 1000 yang memiliki sisa 5 modulo 11 dan 3 modulo 13.

```
>t=1:1000; nonzeros(mod(t^2,11)==5 && mod(t^2,13)==3)
```

```
[4,  48,  95,  139,  147,  191,  238,  282,  290,  334,  381,  425,
433,  477,  524,  568,  576,  620,  667,  711,  719,  763,  810,  854,
862,  906,  953,  997]
```

EMT tidak sepenuhnya efektif untuk perhitungan bilangan bulat. Ia menggunakan titik koma presisi ganda secara internal. Namun, seringkali sangat berguna.

Kita dapat memeriksa apakah sebuah bilangan adalah prima. Mari kita cari tahu berapa banyak kuadrat ditambah 1 yang merupakan bilangan prima.

```
>t=1:1000; length(nonzeros(isprime(t^2+1)))
```

```
112
```

Fungsi nonzeros() hanya berfungsi untuk vektor. Untuk matriks, ada mnonzeros().

```
>seed(2); A=random(3,4)//seed untuk menetapkan angka-angka acak
```

```
0.765761      0.401188      0.406347      0.267829
 0.13673      0.390567      0.495975      0.952814
0.548138      0.006085      0.444255      0.539246
```

Ini mengembalikan indeks dari elemen-elemen yang bukan nol.

```
>k=mnonzeros(A<0.4) //indeks elemen2 A yang kurang dari 0,4
```

```
      1             4
      2             1
      2             2
      3             2
```

Indeks-indeks ini dapat digunakan untuk mengatur elemen-elemen ke suatu nilai tertentu.

```
>mset(A,k,0) //mengganti elemen2 suatu matriks pada indeks tertentu
```

```
0.765761      0.401188      0.406347             0
       0             0      0.495975      0.952814
0.548138             0      0.444255      0.539246
```

Fungsi mset() juga dapat mengatur elemen-elemen pada indeks-indeks tersebut ke entri-entri dari matriks lainnya.

```
>mset(A,k,-random(size(A)))
```

```
 0.765761      0.401188      0.406347     -0.126917
-0.122404     -0.691673      0.495975      0.952814
 0.548138     -0.483902      0.444255      0.539246
```

Dan memungkinkan untuk mendapatkan elemen-elemen dalam bentuk vektor.

```
>mget(A,k)//mendapatkan elemen-elemen dari A dengan indeks k
```

```
[0.267829,  0.13673,  0.390567,  0.006085]
```

Fungsi lain yang berguna adalah extrema, yang mengembalikan nilai minimal dan maksimal dalam setiap baris matriks serta posisinya.

```
>ex=extrema(A)
```

```
      0.267829            4      0.765761            1
       0.13673            1      0.952814            4
      0.006085            2      0.548138            1
```

Kita dapat menggunakan ini untuk mengekstrak nilai maksimal dalam setiap baris.

```
>ex[,3]'
```

```
[0.765761,  0.952814,  0.548138]
```

Ini, tentu saja, sama dengan fungsi max().

```
>max(A)'
```

```
[0.765761,  0.952814,  0.548138]
```

Tapi dengan mget(), kita bisa mengambil indeks dan menggunakan informasi ini untuk mengambil elemen-elemen pada posisi yang sama dari matriks lain.

```
>j=(1:rows(A))'|ex[,4], mget(-A,j)
```

```
             1             1
             2             4
             3             1
[-0.765761,  -0.952814,  -0.548138]
```

Contoh lain

```
>(2:5)*(2:5)'
```

```
            4             6             8            10
            6             9            12            15
            8            12            16            20
           10            15            20            25
```

```
>(2:5).(2:5)'
```

```
54
```

```
>sum((2:5)*(2:5))
```

```
54
```

```
>(1:12)>5/2
```

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

```
>sum((1:12)>5/2)
```

```
10
```

```
>k=(2:20); k==11
```

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

```
>nonzeros(k>11)
```

```
[11,  12,  13,  14,  15,  16,  17,  18,  19]
```

```
>k[nonzeros(k<11)]
```

```
[2,  3,  4,  5,  6,  7,  8,  9,  10]
```

```
>r=1:200; nonzeros(mod(r,2)==1)//mencari daftar bilangan ganjil dari 1 samp
```

```
[1,  3,  5,  7,  9,  11,  13,  15,  17,  19,  21,  23,  25,  27,  29,
31,  33,  35,  37,  39,  41,  43,  45,  47,  49,  51,  53,  55,  57,
59,  61,  63,  65,  67,  69,  71,  73,  75,  77,  79,  81,  83,  85,
87,  89,  91,  93,  95,  97,  99,  101,  103,  105,  107,  109,  111,
113,  115,  117,  119,  121,  123,  125,  127,  129,  131,  133,  135,
137,  139,  141,  143,  145,  147,  149,  151,  153,  155,  157,  159,
161,  163,  165,  167,  169,  171,  173,  175,  177,  179,  181,  183,
185,  187,  189,  191,  193,  195,  197,  199]
```

```
>g=mnonzeros(D<4)
```

```
              1             1
              1             2
              2             1
              2             2
```

```
>mget(E,g)
```

```
[3,  2,  4,  3]
```

```
>ex=extrema(C)
```

```
              2             1             8             4
              3             1            12             4
```

Fungsi Matriks Lainnya (Membangun Matriks)

Untuk membangun sebuah matriks, kita dapat menumpuk satu matriks di atas yang lain. Jika keduanya tidak memiliki jumlah kolom yang sama, yang lebih pendek akan diisi dengan 0.

```
>v=1:3; v_v
```

```
              1             2             3
              1             2             3
```

Demikian pula, kita dapat melampirkan sebuah matriks ke samping yang lain, jika keduanya memiliki jumlah baris yang sama.

```
>A=random(3,4); A|v'
```

```
       0.032444     0.0534171      0.595713      0.564454             1
        0.83916      0.175552      0.396988       0.83514             2
      0.0257573      0.658585      0.629832      0.770895             3
```

Jika keduanya tidak memiliki jumlah baris yang sama, matriks yang lebih pendek akan diisi dengan 0.

Ada pengecualian untuk aturan ini. Sebuah bilangan real yang dilampirkan ke sebuah matriks akan digunakan sebagai kolom yang diisi dengan bilangan real tersebut.

```
>A|1
```

```
       0.032444     0.0534171      0.595713      0.564454             1
        0.83916      0.175552      0.396988       0.83514             1
      0.0257573      0.658585      0.629832      0.770895             1
```

Mungkin membuat matriks dari vektor baris dan kolom.

```
>[v;v]
```

```
              1             2             3
              1             2             3
```

```
>[v',v']
```

```
              1             1
              2             2
              3             3
```

Tujuan utamanya adalah untuk menginterpretasikan sebuah vektor dari ekspresi sebagai vektor kolom.

```
>"[x,x^2]"(v')
```

```
              1             1
              2             4
              3             9
```

Untuk mendapatkan ukuran matriks A, kita dapat menggunakan fungsi-fungsi berikut.

```
>C=zeros(2,4), rows(C), cols(C), size(C), length(C)
```

```
              0             0             0             0
              0             0             0             0
  2
  4
  [2,  4]
  4
```

Untuk vektor, ada fungsi length().

```
>length(2:10)
```

```
  9
```

Ada banyak fungsi lain yang menghasilkan matriks.

```
>ones(2,2)
```

```
              1             1
              1             1
```

Ini juga dapat digunakan dengan satu parameter. Untuk mendapatkan vektor dengan angka selain 1, gunakan yang berikut.

```
>ones(5)*6
```

```
  [6,  6,  6,  6,  6]
```

Juga, matriks dari angka-angka acak dapat dihasilkan dengan random (distribusi seragam) atau normal (distribusi Gaussian).

```
>random(2,2)
```

```
      0.66566      0.831835
        0.977      0.544258
```

Berikut adalah fungsi lain yang berguna, yang mengubah struktur elemen-elemen matriks menjadi matriks lain.

```
>redim(1:9,3,3) // menyusun elemen2 1, 2, 3, ..., 9 ke bentuk matriks 3x3
```

```
            1             2             3
            4             5             6
            7             8             9
```

Dengan fungsi berikut, kita dapat menggunakan ini dan fungsi dup untuk menulis fungsi rep() yang mengulang sebuah vektor sebanyak n kali.

```
>function rep(v,n) := redim(dup(v,n),1,n*cols(v))
```

Mari kita uji.

```
>rep(1:3,5)
```

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

Fungsi multdup() menggandakan elemen-elemen dari vektor.

```
>multdup(1:3,5), multdup(1:3,[2,3,2])
```

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

Fungsi flipx() dan flipy() membalik urutan baris atau kolom matriks. Dengan kata lain, fungsi flipx() melakukan pembalikan horizontal.

```
>flipx(1:5) //membalik elemen2 vektor baris
```

```
  [5,  4,  3,  2,  1]
```

Untuk rotasi, Euler memiliki rotleft() dan rotright().

```
>rotleft(1:5) // memutar elemen2 vektor baris
```

```
[2,  3,  4,  5,  1]
```

Fungsi khusus adalah drop(v, i), yang menghapus elemen-elemen dengan indeks-indeks dalam i dari vektor v.

```
>drop(10:20,3)
```

```
[10,  11,  13,  14,  15,  16,  17,  18,  19,  20]
```

Perhatikan bahwa vektor i dalam drop(v,i) mengacu pada indeks-indeks elemen-elemen dalam v, bukan nilai-nilai elemen. Jika Anda ingin menghapus elemen-elemen, Anda perlu menemukan elemen-elemen tersebut terlebih dahulu. Fungsi indexof(v, x) dapat digunakan untuk menemukan elemen-elemen x dalam vektor yang telah diurutkan.

```
>v=primes(50), i=indexof(v,10:20), drop(v,i)
```

```
[2,  3,  5,  7,  11,  13,  17,  19,  23,  29,  31,  37,  41,  43,  47]
[0,  5,  0,  6,  0,  0,  0,  7,  0,  8,  0]
[2,  3,  5,  7,  23,  29,  31,  37,  41,  43,  47]
```

Catatan tambahan:
-indexof digunakan untuk menemukan kemunculan pertama dari x dalam vektor v
-drop digunakan untuk menghapus elemen-elemen i dari vektor baris v

Seperti yang Anda lihat, tidak masalah jika menyertakan indeks-indeks di luar jangkauan (seperti 0), indeks ganda, atau indeks yang tidak terurut.

```
>drop(1:10,shuffle([0,0,5,5,7,12,12]))
```

```
[1,  2,  3,  4,  6,  8,  9,  10]
```

Ada beberapa fungsi khusus untuk mengatur diagonal atau menghasilkan matriks diagonal.

Kita mulai dengan matriks identitas.

```
>A=id(5) // matriks identitas 5x5
```

```
1    0    0    0    0
0    1    0    0    0
0    0    1    0    0
0    0    0    1    0
0    0    0    0    1
```

Kemudian kita mengatur diagonal bawah (-1) menjadi 1:4.

```
>setdiag(A,-1,1:4) //mengganti diagonal di bawah diagonal utama
```

```
            1             0             0             0             0
            1             1             0             0             0
            0             2             1             0             0
            0             0             3             1             0
            0             0             0             4             1
```

Perhatikan bahwa kita tidak mengubah matriks A. Kita mendapatkan matriks baru sebagai hasil dari setdiag().

Berikut adalah sebuah fungsi yang mengembalikan matriks tri-diagonal.

```
>function tridiag (n,a,b,c) := setdiag(setdiag(b*id(n),1,c),-1,a); ...
>tridiag(5,1,2,3)
```

```
            2             3             0             0             0
            1             2             3             0             0
            0             1             2             3             0
            0             0             1             2             3
            0             0             0             1             2
```

Diagonal dari sebuah matriks juga dapat diekstrak dari matriks itu sendiri. Untuk mendemonstrasikannya, kita merestrukturisasi vektor 1:9 menjadi matriks 3x3.

```
>A=redim(1:9,3,3)
```

```
            1             2             3
            4             5             6
            7             8             9
```

Sekarang kita dapat mengekstrak diagonalnya.

```
>d=getdiag(A,0)
```

```
  [1,   5,   9]
```

Contohnya, kita dapat membagi matriks dengan diagonalnya. Bahasa matriks akan mengurus agar vektor kolom d diterapkan pada matriks baris demi baris.

```
>fraction A/d'
```

```
          1           2           3
        4/5           1         6/5
        7/9         8/9           1
```

Contoh lain

```
>w=4:6; w_w
```

```
              4             5             6
              4             5             6
```

```
>W=random(3,4); W|w'
```

```
       0.208566      0.220144      0.855399     0.0288546             4
       0.259286      0.181379      0.293642      0.791497             5
      0.0155055      0.312754      0.381387      0.875381             6
```

```
>[w; w]
```

```
              4             5             6
              4             5             6
```

```
>[w',w']
```

```
              4             4
              5             5
              6             6
```

```
>"[x,x^3]"(w')
```

```
              4            64
              5           125
              6           216
```

```
>M=zeros(3,7); rows(M), cols(M), size(M), length(M)
```

```
3
7
[3,  7]
7
```

```
>redim(1:4,2,2)
```

```
            1             2
            3             4
```

```
>rep(2:4, 7)
```

```
[2,  3,  4,  2,  3,  4,  2,  3,  4,  2,  3,  4,  2,  3,  4,  2,  3,  4,
2,  3,  4]
```

```
>multdup(2:4,7), multdup(2:4, [2])
```

```
[2,  2,  2,  2,  2,  2,  2,  3,  3,  3,  3,  3,  3,  3,  4,  4,  4,  4,
4,  4,  4]
[2,  2,  3,  3,  4,  4]
```

```
>flipx(999:1005)
```

```
[1005,  1004,  1003,  1002,  1001,  1000,  999]
```

```
>rotleft(999:1005)
```

```
[1000,  1001,  1002,  1003,  1004,  1005,  999]
```

```
>drop(20:30,3)
```

```
[20,  21,  23,  24,  25,  26,  27,  28,  29,  30]
```

```
>K=id(3)
```

```
            1            0            0
            0            1            0
            0            0            1
```

```
>setdiag(K,-1,1:4)
```

```
            1            0            0
            1            1            0
            0            2            1
```

```
>M=redim(1:9,3,3)
```

```
            1            2            3
            4            5            6
            7            8            9
```

```
>m=getdiag(M,0)
```

```
[1,  5,  9]
```

```
>fraction M/m'
```

```
        1         2         3
      4/5         1       6/5
      7/9       8/9         1
```

Vektorisasi

Hampir semua fungsi dalam Euler berfungsi juga untuk input matriks dan vektor, bila ini masuk akal.

Sebagai contoh, fungsi sqrt() menghitung akar kuadrat dari semua elemen vektor atau matriks.

```
>sqrt(1:3)
```

```
[1,  1.41421,  1.73205]
```

Jadi, Anda dapat dengan mudah membuat tabel nilai. Ini adalah salah satu cara untuk membuat grafik fungsi (alternatifnya menggunakan ungkapan).

```
>x=1:0.01:5; y=log(x)/x^2; // terlalu panjang untuk ditampikan
```

Dengan ini dan operator titik dua a:delta:b, vektor nilai dari fungsi dapat dibuat dengan mudah.

Pada contoh berikut, kami menghasilkan vektor nilai t[i] dengan selang 0,1 dari -1 hingga 1. Kemudian kami menghasilkan vektor nilai dari fungsi

$$s = t^3 - t$$

```
>t=-1:0.1:1; s=t^3-t
```

```
[0,  0.171,  0.288,  0.357,  0.384,  0.375,  0.336,  0.273,  0.192,
0.099,  0,  -0.099,  -0.192,  -0.273,  -0.336,  -0.375,  -0.384,
-0.357,  -0.288,  -0.171,  0]
```

EMT memperluas operator untuk skalar, vektor, dan matriks dengan cara yang jelas.

Contohnya, perkalian antara vektor kolom dan vektor baris akan diperluas menjadi matriks jika operator diterapkan. Dalam contoh berikut, v' adalah vektor transpos (vektor kolom).

```
>shortest (1:5)*(1:5)'
```

```
       1       2       3       4       5
       2       4       6       8      10
       3       6       9      12      15
       4       8      12      16      20
       5      10      15      20      25
```

Perlu diperhatikan bahwa ini sangat berbeda dari perkalian matriks. Perkalian matriks ditandai dengan titik "." dalam EMT.

```
>(1:5).(1:5)'
```

```
55
```

Secara default, vektor baris akan dicetak dalam format yang ringkas.

```
>[1,2,3,4]
```

```
[1,  2,  3,  4]
```

Untuk matriks, operator khusus "." menunjukkan perkalian matriks, dan A' menunjukkan transpose. Matriks 1x1 dapat digunakan sama seperti angka riil.

```
>v:=[1,2]; v.v', %^2
```

```
5
25
```

Untuk melakukan transpose pada sebuah matriks, kita menggunakan tanda apostrof (').

```
>v=1:4; v'
```

```
              1
              2
              3
              4
```

Jadi, kita dapat menghitung perkalian matriks A dengan vektor b.

```
>A=[1,2,3,4;5,6,7,8]; A.v'
```

```
             30
             70
```

Perlu diingat bahwa v tetap merupakan vektor baris. Jadi, v'.v berbeda dari v.v'.

```
>v'.v
```

```
              1              2              3              4
              2              4              6              8
              3              6              9             12
              4              8             12             16
```

v.v' menghitung norma dari v kuadrat untuk vektor baris v. Hasilnya adalah vektor 1x1, yang berfungsi seperti bilangan riil.

```
>v.v'
```

```
30
```

Ada juga fungsi norma (bersama dengan banyak fungsi lain dari Aljabar Linear).

```
>norm(v)^2
```

```
30
```

Operator dan fungsi mengikuti bahasa matriks Euler.

Berikut adalah ringkasan aturan-aturannya:

- Fungsi yang diterapkan pada vektor atau matriks diterapkan pada setiap elemen.

- Operator yang beroperasi pada dua matriks dengan ukuran yang sama diterapkan secara berpasangan pada elemen-elemen matriks tersebut.

- Jika dua matriks memiliki dimensi yang berbeda, keduanya diperluas dengan cara yang masuk akal, sehingga memiliki ukuran yang sama.

Misalnya, nilai skalar dikalikan dengan vektor mengalikan nilai tersebut dengan setiap elemen vektor. Atau matriks dikalikan dengan vektor (dengan *, bukan .) akan memperluas vektor ke ukuran matriks dengan menggandakannya.

Berikut adalah contoh sederhana dengan operator ^.

```
>[1,2,3]^2
```

```
[1,   4,   9]
```

Berikut adalah kasus yang lebih rumit. Sebuah vektor baris dikali dengan vektor kolom akan memperluas keduanya dengan menggandakannya.

```
>v:=[1,2,3]; v*v'
```

```
              1              2              3
              2              4              6
              3              6              9
```

Perlu diperhatikan bahwa produk skalar menggunakan perkalian matriks, bukan *!

```
>v.v'
```

```
14
```

Ada banyak fungsi untuk matriks. Berikut adalah daftar singkat. Anda sebaiknya merujuk ke dokumentasi untuk informasi lebih lanjut tentang perintah-perintah ini.

- sum, prod menghitung jumlah dan produk dari baris-baris
- cumsum, cumprod melakukan hal yang sama secara kumulatif
- menghitung nilai-nilai ekstrem dari setiap baris
- extrema mengembalikan vektor dengan informasi ekstremal
- diag(A,i) mengembalikan diagonal ke-i
- setdiag(A,i,v) mengatur diagonal ke-i
- id(n) matriks identitas
- det(A) determinan
- charpoly(A) polinom karakteristik
- eigenvalues(A) nilai-nilai eigen

```
>v*v, sum(v*v), cumsum(v*v)
```

```
[1,  4,  9]
14
[1,  5,  14]
```

Operator titik dua ":" menghasilkan vektor baris dengan jarak yang sama, opsional dengan ukuran langkah.

```
>1:4, 1:2:10
```

```
[1,  2,  3,  4]
[1,  3,  5,  7,  9]
```

Untuk menggabungkan matriks dan vektor, terdapat operator "|" dan "_".

```
>[1,2,3]|[4,5], [1,2,3]_1
```

```
[1,  2,  3,  4,  5]
            1             2             3
            1             1             1
```

Elemen-elemen dari sebuah matriks dirujuk dengan "A[i,j]".

```
>A:=[1,2,3;4,5,6;7,8,9]; A[2,3]
```

```
6
```

Untuk vektor baris atau vektor kolom, v[i] adalah elemen ke-i dari vektor tersebut. Untuk matriks, ini mengembalikan seluruh baris ke-i dari matriks.

```
>v:=[2,4,6,8]; v[3], A[3]
```

```
6
[7,  8,  9]
```

Indeks juga bisa berupa vektor baris dari indeks. ":" menunjukkan semua indeks.

```
>v[1:2], A[:,2]
```

```
[2,  4]
            2
            5
            8
```

Bentuk singkat untuk ":" adalah dengan menghilangkan indeks sepenuhnya.

```
>A[,2:3]
```

```
            2             3
            5             6
            8             9
```

Untuk keperluan vektorisasi, elemen-elemen dari sebuah matriks dapat diakses seolah-olah mereka adalah vektor.

```
>A{4}
```

```
4
```

Sebuah matriks juga dapat "diratakan" (flattened), menggunakan fungsi redim(). Ini diimplementasikan dalam fungsi flatten().

```
>redim(A,1,prod(size(A))), flatten(A)
```

```
[1,  2,  3,  4,  5,  6,  7,  8,  9]
[1,  2,  3,  4,  5,  6,  7,  8,  9]
```

Untuk menggunakan matriks untuk tabel, mari kembalikan ke format default dan hitung tabel nilai-nilai sinus dan kosinus. Perlu diingat bahwa sudutnya dalam radian secara default.

```
>defformat; w=0°:45°:360°; w=w'; deg(w)
```

```
              0
             45
             90
            135
            180
            225
            270
            315
            360
```

Sekarang kita akan menambahkan kolom-kolom ke dalam sebuah matriks.

```
>M = deg(w)|w|cos(w)|sin(w)
```

```
              0             0             1             0
             45      0.785398      0.707107      0.707107
             90        1.5708             0             1
            135       2.35619     -0.707107      0.707107
            180       3.14159            -1             0
            225       3.92699     -0.707107     -0.707107
            270       4.71239             0            -1
            315       5.49779      0.707107     -0.707107
            360       6.28319             1             0
```

Dengan menggunakan bahasa matriks, kita dapat menghasilkan beberapa tabel dari beberapa fungsi sekaligus.

Pada contoh berikut, kita menghitung t[j]^i untuk i mulai dari 1 hingga n. Kita mendapatkan sebuah matriks, di mana setiap baris merupakan tabel dari t^i untuk satu nilai i. Artinya, matriks tersebut memiliki elemen-elemen latex: a_{i,j} = t_j^i, \quad 1 \le j \le 101, \quad 1 \le i \le n

Sebuah fungsi yang tidak berfungsi untuk masukan vektor harus "divektorisasi". Ini dapat dicapai dengan kata kunci "map" dalam definisi fungsi. Kemudian fungsi akan dievaluasi untuk setiap elemen dari parameter vektor.

Pengintegrasi numerik integrate() hanya berfungsi untuk batas interval skalar. Jadi kita perlu melakukan vektorisasi terhadapnya.

```
>function map f(x) := integrate("x^x",1,x)
```

Kata kunci "map" melakukan vektorisasi pada fungsi tersebut. Fungsi ini sekarang akan berfungsi untuk vektor-vektor angka.

```
>f([1:5])
```

```
[0,  2.05045,  13.7251,  113.336,  1241.03]
```

Contoh lain

```
>k=2:5
```

```
[2,  3,  4,  5]
```

```
>norm(k)^2
```

```
54
```

```
>k*k, sum(k*k), cumsum(k*k)
```

```
[4,  9,  16,  25]
54
[4,  13,  29,  54]
```

```
>A:=[1,2;4,5;7,8]
```

```
            1             2
            4             5
            7             8
```

```
>redim(A,1,prod(size(A))), flatten(A)
```

```
[1,  2,  4,  5,  7,  8]
[1,  2,  4,  5,  7,  8]
```

```
>defformat; w=0°:15°:180°; w=w'; deg(w)
```

```
             0
            15
            30
            45
            60
            75
            90
           105
           120
           135
           150
           165
           180
```

```
>M = deg(w)|w|cos(w)|sin(w)
```

```
             0             0             1             0
            15      0.261799      0.965926      0.258819
            30      0.523599      0.866025           0.5
            45      0.785398      0.707107      0.707107
            60        1.0472           0.5      0.866025
            75         1.309      0.258819      0.965926
            90        1.5708             0             1
           105        1.8326     -0.258819      0.965926
           120        2.0944          -0.5      0.866025
           135       2.35619     -0.707107      0.707107
           150       2.61799     -0.866025           0.5
           165       2.87979     -0.965926      0.258819
           180       3.14159            -1             0
```

```
>f([2:7])
```

```
[2.05045,  13.7251,  113.336,  1241.03,  17128.1,  284713]
```

Sub-Matriks dan Elemen Matriks

Untuk mengakses elemen matriks, gunakan notasi tanda kurung.

```
>A=[1,2,3;4,5,6;7,8,9], A[2,2]
```

```
                  1                   2                   3
                  4                   5                   6
                  7                   8                   9
  5
```

Kita dapat mengakses seluruh baris dari sebuah matriks.

```
>A[2]
```

```
  [4,  5,  6]
```

Dalam kasus vektor baris atau vektor kolom, ini mengembalikan sebuah elemen dari vektor tersebut.

```
>v=1:3; v[2]
```

```
  2
```

Untuk memastikan Anda mendapatkan baris pertama untuk sebuah matriks 1xn dan matriks mxn, tentukan semua kolom dengan menggunakan indeks kedua yang kosong.

```
>A[2,]
```

```
  [4,  5,  6]
```

Jika indeks adalah vektor indeks, Euler akan mengembalikan baris-baris yang sesuai dari matriks tersebut.
Di sini kita ingin mendapatkan baris pertama dan kedua dari A.

```
>A[[1,2]]
```

```
                  1                   2                   3
                  4                   5                   6
```

Kita bahkan dapat mengurutkan ulang A menggunakan vektor indeks. Untuk menjadi lebih tepat, kita tidak mengubah A di sini, tetapi menghitung versi A yang diurutkan ulang.

```
>A[[3,2,1]]
```

7	8	9
4	5	6
1	2	3

Trik indeks ini juga berfungsi dengan kolom-kolom.
Contoh ini memilih semua baris dari A dan kolom kedua dan ketiga.

```
>A[1:3,2:3]
```

2	3
5	6
8	9

Untuk singkatan, ":" menunjukkan semua indeks baris atau kolom.

```
>A[:,3]
```

3
6
9

Sebagai alternatif, biarkan indeks pertama kosong.

```
>A[,2:3]
```

2	3
5	6
8	9

Kita juga dapat mendapatkan baris terakhir dari A.

```
>A[-1]
```

```
[7,  8,  9]
```

Sekarang mari kita ubah elemen-elemen A dengan memberikan sebuah submatriks dari A ke beberapa nilai. Ini sebenarnya mengubah matriks A yang tersimpan.

```
>A[1,1]=4
```

```
            4             2             3
            4             5             6
            7             8             9
```

Kita juga dapat memberikan sebuah nilai kepada sebuah baris dari A.

```
>A[1]=[-1,-1,-1]
```

```
           -1            -1            -1
            4             5             6
            7             8             9
```

Kita bahkan dapat memberikan nilai kepada sebuah submatriks jika ukurannya sesuai.

```
>A[1:2,1:2]=[5,6;7,8]
```

```
            5             6            -1
            7             8             6
            7             8             9
```

Selain itu, beberapa pintasan juga diperbolehkan.

```
>A[1:2,1:2]=0
```

```
            0             0            -1
            0             0             6
            7             8             9
```

Peringatan: Indeks yang keluar dari batas akan mengembalikan matriks kosong atau pesan kesalahan, tergantung pada pengaturan sistem. Secara default, pesan kesalahan akan ditampilkan. Namun, perlu diingat bahwa indeks negatif dapat digunakan untuk mengakses elemen-elemen matriks dengan menghitung dari akhir.

```
>A[3]
```

```
  [7,  8,  9]
```

Contoh lain

```
>B=[4,4,5;5,4,6;7,8,9], A[3,3]
```

```
              4             4             5
              5             4             6
              7             8             9
   9
```

```
>B[2]
```

```
   [5,   4,   6]
```

```
>B[[3,2,1]]
```

```
              7             8             9
              5             4             6
              4             4             5
```

```
>B[,2:3]
```

```
              4             5
              4             6
              8             9
```

```
>B[1:2,1:2]=0
```

```
              0             0             5
              0             0             6
              7             8             9
```

```
>A[2]
```

```
   [0,   0,   6]
```

Pengurutan dan Pengacakan

Fungsi sort() mengurutkan vektor baris.

```
>sort([5,6,4,8,1,9])
```

```
[1,  4,  5,  6,  8,  9]
```

Seringkali penting untuk mengetahui indeks dari vektor yang sudah diurutkan dalam vektor aslinya. Ini dapat digunakan untuk mengurutkan kembali vektor lain dengan cara yang sama.

Mari kita mengacak sebuah vektor.

```
>v=shuffle(1:10)
```

```
[5,  9,  10,  1,  3,  8,  7,  4,  6,  2]
```

Indeks mengandung urutan yang tepat dari v.

```
>{vs,ind}=sort(v); v[ind]
```

```
[1,  2,  3,  4,  5,  6,  7,  8,  9,  10]
```

Ini juga berfungsi untuk vektor string.

```
>s=["a","d","e","a","aa","e"]
```

```
a
d
e
a
aa
e
```

```
>{ss,ind}=sort(s); ss
```

```
a
a
aa
d
e
e
```

Seperti yang Anda lihat, posisi dari entri ganda agak acak.

```
>ind
```

```
[4,  1,  5,  2,  6,  3]
```

Fungsi unique mengembalikan daftar terurut dari elemen-elemen unik dari sebuah vektor.

```
>intrandom(1,10,10), unique(%)
```

```
[5,  8,  5,  2,  7,  10,  4,  4,  2,  1]
[1,  2,  4,  5,  7,  8,  10]
```

Ini juga berfungsi untuk vektor string.

```
>unique(s)
```

```
a
aa
d
e
```

Aljabar Linear

EMT memiliki banyak fungsi untuk menyelesaikan sistem linear, sistem sparse, atau masalah regresi.

Untuk sistem linear Ax=b, Anda dapat menggunakan algoritma Gauss, matriks invers, atau regresi linear. Operator A\b menggunakan versi algoritma Gauss.

```
>A=[1,2;3,4]; b=[5;6]; A\b
```

```
         -4
        4.5
```

Sebagai contoh lain, kita menghasilkan sebuah matriks berukuran 200x200 dan menjumlahkan semua barisnya. Kemudian kita menyelesaikan Ax=b menggunakan matriks invers. Kita mengukur kesalahan sebagai deviasi maksimum dari semua elemen dari nilai 1, yang tentu saja adalah solusi yang benar.

```
>A=normal(200,200); b=sum(A); longest totalmax(abs(inv(A).b-1))
```

```
8.810729923425242e-13
```

Jika sistem tersebut tidak memiliki solusi, regresi linear akan meminimalkan norma dari kesalahan Ax-b.

```
>A=[1,2,3;4,5,6;7,8,9]
```

```
              1             2             3
              4             5             6
              7             8             9
```

Determinan matriks ini adalah 0.

```
>det(A)
```

```
0
```

Matriks Simbolis

Maxima memiliki matriks simbolis. Tentu saja, Maxima dapat digunakan untuk masalah aljabar linear yang sederhana. Kita dapat mendefinisikan matriks untuk Euler dan Maxima dengan &:=, dan kemudian menggunakannya dalam ekspresi simbolis. Bentuk biasa [...] untuk mendefinisikan matriks dapat digunakan dalam Euler untuk mendefinisikan matriks simbolis.

```
>A &= [a,1,1;1,a,1;1,1,a]; $A
```

$$\begin{pmatrix} a & 1 & 1 \\ 1 & a & 1 \\ 1 & 1 & a \end{pmatrix}$$

```
>$&det(A), $&factor(%)
```

$$a\,\left(a^2-1\right)-2\,a+2$$

$$\left(a-1\right)^2\,\left(a+2\right)$$

```
>$&invert(A) with a=0
```

$$\begin{pmatrix} -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{pmatrix}$$

```
>A &= [1,a;b,2]; $A
```

$$\begin{pmatrix} 1 & a \\ b & 2 \end{pmatrix}$$

Seperti semua variabel simbolis, matriks-matriks ini dapat digunakan dalam ekspresi simbolis lainnya.

```
>$&det(A-x*ident(2)), $&solve(%,x)
```

$$(1 - x)\ (2 - x) - a\, b$$

$$\left[x = \frac{3 - \sqrt{4\, a\, b + 1}}{2}, x = \frac{\sqrt{4\, a\, b + 1} + 3}{2}\right]$$

Nilai-nilai eigen juga dapat dihitung secara otomatis. Hasilnya adalah vektor dengan dua vektor nilai eigen dan multipelitasnya.

```
>$&eigenvalues([a,1;1,a])
```

$$[[a - 1, a + 1]\, , [1, 1]]$$

Untuk mengambil sebuah vektor eigen tertentu, perlu perhatian khusus pada indeksnya.

```
>$&eigenvectors([a,1;1,a]), &%[2][1][1]
```

$$[[[a - 1, a + 1]\, , [1, 1]]\, , [[[1, -1]]\, , [[1, 1]]]]$$

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

Matriks simbolis dapat dievaluasi secara numerik dalam Euler seperti ekspresi simbolis lainnya.

```
>A(a=4,b=5)
```

```
1                    4
5                    2
```

Dalam ekspresi simbolis, gunakan dengan ("with").

```
>$&A with [a=4,b=5]
```

$$\begin{pmatrix} 1 & 4 \\ 5 & 2 \end{pmatrix}$$

Akses ke baris dari matriks simbolis berfungsi sama seperti pada matriks numerik.

```
>$&A[1]
```

$$[1, a]$$

Ekspresi simbolis dapat berisi sebuah penugasan, dan itu mengubah matriks A.

```
>&A[1,1]:=t+1; $&A
```

$$\begin{pmatrix} t+1 & a \\ b & 2 \end{pmatrix}$$

Ada fungsi simbolis dalam Maxima untuk membuat vektor dan matriks. Untuk ini, lihat dokumentasi Maxima atau tutorial tentang Maxima di EMT.

```
>v &= makelist(1/(i+j),i,1,3); $v
```

$$\left[\frac{1}{j+1}, \frac{1}{j+2}, \frac{1}{j+3}\right]$$

```
>B &:= [1,2;3,4]; $B, $&invert(B)
```

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

$$\begin{pmatrix} -2 & 1 \\ \frac{3}{2} & -\frac{1}{2} \end{pmatrix}$$

Hasilnya dapat dievaluasi secara numerik dalam Euler. Untuk informasi lebih lanjut tentang Maxima, lihat pengantar tentang Maxima.

```
>$&invert(B)()
```

```
            -2              1
           1.5           -0.5
```

Euler juga memiliki fungsi yang kuat yaitu xinv(), yang melakukan upaya lebih besar dan memberikan hasil yang lebih akurat.

Perhatikan bahwa dengan &:= matriks B telah didefinisikan sebagai simbolik dalam ekspresi simbolik dan sebagai numerik dalam ekspresi numerik. Jadi kita dapat menggunakannya di sini.

```
>longest B.xinv(B)
```

```
                       1                       0
                       0                       1
```

Contohnya, nilai-nilai eigen dari A dapat dihitung secara numerik.

```
>A=[1,2,3;4,5,6;7,8,9]; real(eigenvalues(A))
```

```
  [16.1168,  -1.11684,  0]
```

Atau secara simbolis. Lihat tutorial tentang Maxima untuk detailnya.

```
>$&eigenvalues(@A)
```

$$\left[\left[\frac{15-3\,\sqrt{33}}{2}, \frac{3\,\sqrt{33}+15}{2}, 0\right], [1,1,1]\right]$$

Nilai Numerik dalam Ekspresi Simbolis

Sebuah ekspresi simbolis hanyalah sebuah string yang berisi ekspresi. Jika kita ingin mendefinisikan nilai baik untuk ekspresi simbolis maupun ekspresi numerik, kita harus menggunakan "&:=".

```
>A&:= [1,pi;4,5]
```

```
                  1        3.14159
                  4              5
```

Masih ada perbedaan antara bentuk numerik dan bentuk simbolis. Ketika mentransfer matriks ke bentuk simbolis, akan digunakan pendekatan pecahan untuk bilangan real.

```
>$&A
```

$$\begin{pmatrix} 1 & \frac{1146408}{364913} \\ 4 & 5 \end{pmatrix}$$

Untuk menghindari hal ini, ada fungsi "mxmset(variable)".

```
>mxmset(A); $&A
```

$$\begin{pmatrix} 1 & 3.141592653589793 \\ 4 & 5 \end{pmatrix}$$

Maxima juga dapat melakukan perhitungan dengan angka desimal, bahkan dengan angka desimal besar dengan 32 digit. Namun, evaluasinya akan lebih lambat.

```
>$&bfloat(sqrt(2)), $&float(sqrt(2))
```

$$1.4142135623730950488016887242097_B \times 10^0$$

$$1.414213562373095$$

Presisi angka desimal besar dapat diubah.

```
>&fpprec:=100; &bfloat(pi)
```

```
          3.14159265358979323846264338327950288419716939937510582097494\
 4592307816406286208998628034825342117068b0
```

Variabel numerik dapat digunakan dalam ekspresi simbolis menggunakan "@var".

Perlu diingat bahwa ini hanya diperlukan jika variabel tersebut telah didefinisikan dengan ":=" atau "=" sebagai variabel numerik.

```
>B:=[1,pi;3,4]; $&det(@B)
```

$$-5.424777960769379$$

Demo - Tingkat Bunga

Di bawah ini, kita menggunakan Euler Math Toolbox (EMT) untuk menghitung tingkat bunga. Kita melakukannya secara numerik dan simbolis untuk menunjukkan bagaimana Euler dapat digunakan untuk menyelesaikan masalah dunia nyata.

Misalkan Anda memiliki modal awal sebesar 5000 (katakanlah dalam dolar).

```
>K=5000
```

```
5000
```

Sekarang kita asumsikan tingkat bunga sebesar 3% per tahun. Mari tambahkan satu tingkat bunga sederhana dan hitung hasilnya.

```
>K*1.03
```

```
5150
```

Euler akan memahami sintaks berikut juga.

```
>K+K*3%
```

```
5150
```

Namun, lebih mudah menggunakan faktor.

```
>q=1+3%, K*q
```

```
1.03
5150
```

Selama 10 tahun, kita dapat dengan mudah mengalikan faktor-faktor tersebut dan mendapatkan nilai akhir dengan tingkat bunga majemuk.

```
>K*q^10
```

```
6719.58189672
```

Untuk keperluan kita, kita dapat mengatur formatnya menjadi 2 digit setelah tanda desimal.

```
>format(12,2); K*q^10
```

```
      6719.58
```

Mari cetak itu dibulatkan menjadi 2 digit dalam sebuah kalimat lengkap.

```
>"Starting from " + K + "$ you get " + round(K*q^10,2) + "$."
```

```
Starting from 5000$ you get 6719.58$.
```

Bagaimana jika kita ingin mengetahui hasil-hasil perantara dari tahun 1 hingga tahun 9? Untuk ini, bahasa matriks Euler sangat membantu. Anda tidak perlu menulis sebuah perulangan, tetapi cukup masukkan:

```
>K*q^(0:10)
```

```
Real 1 x 11 matrix

      5000.00      5150.00      5304.50      5463.64      ...
```

Bagaimana cara kerja keajaiban ini? Pertama, ekspresi 0:10 mengembalikan vektor bilangan bulat.

```
>short 0:10
```

```
[0,  1,  2,  3,  4,  5,  6,  7,  8,  9,  10]
```

Kemudian semua operator dan fungsi dalam Euler dapat diterapkan pada elemen-elemen vektor secara berurutan. Jadi,

```
>short q^(0:10)
```

```
[1,  1.03,  1.0609,  1.0927,  1.1255,  1.1593,  1.1941,  1.2299,
1.2668,  1.3048,  1.3439]
```

adalah vektor faktor dari q^0 hingga q^10. Ini dikalikan dengan K, dan kita mendapatkan vektor nilai.

```
>VK=K*q^(0:10);
```

Tentu saja, cara yang realistis untuk menghitung tingkat bunga ini adalah dengan membulatkan ke sen terdekat setelah setiap tahun. Mari tambahkan sebuah fungsi untuk ini.

```
>function oneyear (K) := round(K*q,2)
```

Mari membandingkan dua hasil, dengan dan tanpa pembulatan.

```
>longest oneyear(1234.57), longest 1234.57*q
```

```
                    1271.61
                  1271.6071
```

Sekarang tidak ada rumus sederhana untuk tahun ke-n, dan kita harus melakukan perulangan selama beberapa tahun. Euler menyediakan banyak solusi untuk ini.

Cara termudah adalah fungsi iterate, yang mengulangi suatu fungsi yang diberikan sejumlah kali.

```
>VKr=iterate("oneyear",5000,10)
```

```
Real 1 x 11 matrix

    5000.00     5150.00     5304.50     5463.64     ...
```

Kita dapat mencetaknya dengan cara yang ramah, menggunakan format dengan angka desimal tetap.

```
>VKr'
```

```
5000.00
5150.00
5304.50
5463.64
5627.55
5796.38
5970.27
6149.38
6333.86
6523.88
6719.60
```

Untuk mendapatkan elemen tertentu dari vektor, kita menggunakan indeks dalam kurung siku.

```
>VKr[2], VKr[1:3]
```

```
5150.00
5000.00      5150.00      5304.50
```

Mengejutkan, kita juga dapat menggunakan vektor indeks. Ingatlah bahwa 1:3 menghasilkan vektor [1,2,3].

Mari bandingkan elemen terakhir dari nilai-nilai yang dibulatkan dengan nilai-nilai lengkap.

```
>VKr[-1], VK[-1]
```

```
6719.60
6719.58
```

Perbedaannya sangat kecil.

Menyelesaikan Persamaan

Sekarang kita akan menggunakan fungsi yang lebih canggih, yang menambahkan jumlah uang tertentu setiap tahunnya.

```
>function onepay (K) := K*q+R
```

Kita tidak perlu menentukan q atau R untuk definisi fungsi. Hanya jika kita menjalankan perintah, kita harus menentukan nilai-nilai ini. Kita memilih R=200.

```
>R=200; iterate("onepay",5000,10)
```

```
  Real 1 x 11 matrix

      5000.00     5350.00     5710.50     6081.82     ...
```

Bagaimana jika kita menghapus jumlah yang sama setiap tahun?

```
>R=-200; iterate("onepay",5000,10)
```

```
  Real 1 x 11 matrix

      5000.00     4950.00     4898.50     4845.45     ...
```

Kita bisa melihat bahwa jumlah uangnya berkurang. Tentu saja, jika kita hanya mendapatkan 150 dari bunga pada tahun pertama, tetapi mengambil 200, kita akan kehilangan uang setiap tahun.

Bagaimana kita bisa menentukan berapa tahun uangnya akan habis? Kita harus menulis perulangan untuk ini. Cara termudah adalah dengan mengulangi cukup lama.

```
>VKR=iterate("onepay",5000,50)
```

```
  Real 1 x 51 matrix

      5000.00     4950.00     4898.50     4845.45     ...
```

Dengan menggunakan bahasa matriks, kita dapat menentukan nilai negatif pertama dengan cara berikut.

```
>min(nonzeros(VKR<0))
```

```
        48.00
```

Alasan untuk ini adalah bahwa nonzeros(VKR<0) mengembalikan vektor indeks i, di mana VKR[i]<0, dan min menghitung indeks minimal.

Karena vektor selalu dimulai dengan indeks 1, jawabannya adalah 47 tahun.

Fungsi iterate() memiliki satu trik lagi. Itu bisa mengambil kondisi akhir sebagai argumen. Kemudian itu akan mengembalikan nilai dan jumlah iterasi.

```
>{x,n}=iterate("onepay",5000,till="x<0"); x, n,
```

```
-19.83
 47.00
```

Mari mencoba menjawab pertanyaan yang lebih ambigu. Misalkan kita tahu bahwa nilai adalah 0 setelah 50 tahun. Berapa tingkat bunga yang akan diterapkan?

Ini adalah pertanyaan yang hanya bisa dijawab secara numerik. Di bawah ini, kita akan mendapatkan rumus yang diperlukan. Kemudian Anda akan melihat bahwa tidak ada rumus yang mudah untuk tingkat bunga. Tetapi untuk saat ini, kita akan mencari solusi numerik.

Langkah pertama adalah mendefinisikan sebuah fungsi yang melakukan iterasi sebanyak n kali. Kita akan menambahkan semua parameter ke fungsi ini.

```
>function f(K,R,P,n) := iterate("x*(1+P/100)+R",K,n;P,R)[-1]
```

Iterasinya sama seperti di atas

$$x_{n+1} = x_n \cdot \left(1 + \frac{P}{100}\right) + R$$

Tetapi kita tidak lagi menggunakan nilai global dari R dalam ekspresi kita. Fungsi-fungsi seperti iterate() memiliki trik khusus di Euler. Anda dapat melewati nilai-nilai variabel dalam ekspresi sebagai parameter titik koma. Dalam hal ini P dan R.

Selain itu, kita hanya tertarik pada nilai terakhir. Jadi kita mengambil indeks [-1].

Mari kita coba uji coba.

```
>f(5000,-200,3,47)
```

```
-19.83
```

Sekarang kita dapat menyelesaikan masalah kita.

```
>solve("f(5000,-200,x,50)",3)
```

```
3.15
```

Rutinitas solve menyelesaikan ekspresi=0 untuk variabel x. Jawabannya adalah 3,15% per tahun. Kita mengambil nilai awal 3% untuk algoritma. Fungsi solve() selalu memerlukan nilai awal.

Kita dapat menggunakan fungsi yang sama untuk menyelesaikan pertanyaan berikut: Berapa banyak yang dapat kita ambil setiap tahun sehingga modal awal habis setelah 20 tahun dengan asumsi tingkat bunga 3% per tahun.

```
>solve("f(5000,x,3,20)",-200)
```

```
-336.08
```

Perhatikan bahwa Anda tidak dapat menyelesaikan untuk jumlah tahun, karena fungsi kami mengasumsikan n sebagai nilai bulat.

Solusi Simbolik untuk Masalah Tingkat Bunga

Kita dapat menggunakan bagian simbolik dari Euler untuk mempelajari masalah ini. Pertama, kita mendefinisikan fungsi onepay() secara simbolis.

```
>function op(K) &= K*q+R; $&op(K)
```

$$R + q\,K$$

Sekarang kita dapat mengulanginya.

```
>$&op(op(op(op(K)))), $&expand(%)
```

$$q\,\left(q\,\left(q\,\left(R+q\,K\right)+R\right)+R\right)+R$$

$$q^3\,R+q^2\,R+q\,R+R+q^4\,K$$

Kita melihat sebuah pola. Setelah n periode kita memiliki

$$K_n = q^n K + R(1 + q + \ldots + q^{n-1}) = q^n K + \frac{q^n - 1}{q - 1} R$$

Rumus ini adalah rumus untuk jumlah geometri, yang dikenal oleh Maxima.

```
>&sum(q^k,k,0,n-1); $& % = ev(%,simpsum)
```

$$\sum_{k=0}^{n-1} q^k = \frac{q^n - 1}{q - 1}$$

Ini sedikit rumit. Penjumlahan dievaluasi dengan pengaturan "simpsum" untuk menguranginya menjadi pecahan.

Mari buat sebuah fungsi untuk ini.

```
>function fs(K,R,P,n) &= (1+P/100)^n*K + ((1+P/100)^n-1)/(P/100)*R; $&fs(K,
```

$$\frac{100\,\left(\left(\frac{P}{100}+1\right)^n-1\right)\,R}{P}+K\,\left(\frac{P}{100}+1\right)^n$$

Fungsi ini melakukan hal yang sama dengan fungsi f sebelumnya. Tetapi lebih efektif.

```
>longest f(5000,-200,3,47), longest fs(5000,-200,3,47)
```

```
      -19.82504734650985
      -19.82504734652684
```

Sekarang kita bisa menggunakannya untuk menanyakan waktu n. Kapan modal kita habis? Tebakan awal kita adalah 30 tahun.

```
>solve("fs(5000,-330,3,x)",30)
```

```
      20.51
```

Jawaban ini mengatakan bahwa uang akan habis setelah 21 tahun.

Kita juga bisa menggunakan sisi simbolik Euler untuk menghitung rumus-rumus pembayaran.

Misalkan kita mendapatkan pinjaman sebesar K, dan membayar n pembayaran sebesar R (dimulai setelah tahun pertama) dengan sisa utang sebesar Kn (pada saat pembayaran terakhir). Rumus ini jelas:

```
>equ &= fs(K,R,P,n)=Kn; $&equ
```

$$\frac{100\,\left(\left(\frac{P}{100}+1\right)^n-1\right)\,R}{P}+K\,\left(\frac{P}{100}+1\right)^n=Kn$$

Biasanya rumus ini dinyatakan dalam bentuk

$$i = \frac{P}{100}$$

```
>equ &= (equ with P=100*i); $&equ
```

$$\frac{\left((i+1)^n-1\right)\,R}{i}+(i+1)^n\,K=Kn$$

Kita dapat menyelesaikan untuk tingkat R secara simbolis.

```
>$&solve(equ,R)
```

$$\left[R=\frac{i\,Kn-i\,(i+1)^n\,K}{(i+1)^n-1}\right]$$

Seperti yang dapat Anda lihat dari rumusnya, fungsi ini menghasilkan kesalahan titik desimal untuk i=0. Namun, Euler tetap memplotnya.

Tentu saja, kita memiliki batasan berikut.

```
>$&limit(R(5000,0,x,10),x,0)
```

$$\lim_{x\rightarrow 0}{R\left(5000 , 0 , x , 10\right)}$$

Jelas, tanpa bunga, kita harus membayar kembali 10 pembayaran sebesar 500.

Persamaan ini juga dapat diselesaikan untuk n. Terlihat lebih bagus jika kita melakukan beberapa penyederhanaan padanya.

```
>fn &= solve(equ,n) | ratsimp; $&fn
```

$$\left[n=\frac{\log \left(\frac{R+i\,Kn}{R+i\,K}\right)}{\log (i+1)}\right]$$

SOAL DARI PDF ALGEBRA EXCERCISES ** R.2

Sederhanakan!
Nomor 1

$$(\frac{24a^{10}b^{-8}c^7}{12a^6b^{-3}c^5})^{-5}$$

jawaban manual:

$$(\frac{24a^{10}b^{-8}c^7}{12a^6b^{-3}c^5})^{-5} = (\frac{2a^4c^2}{b^5})^{-5} =$$

$$(\frac{b^5}{2a^4c^2})^5 = \frac{b^{25}}{32a^{20}c^{10}}$$

jawaban emt:

```
>$ ((24*(a^(10))*(b^(-8))*(c^7))/(12*(a^6)*(b^(-3))*c^5))^(-5)
```

Nomor 2

$$(\frac{125p^{12}q^{-14}r^{22}}{25p^8q^6r^{-15}})^{-4}$$

```
>$ ((125*(p^(12))*(q^(-14))*(r^(22)))/(25*(p^8)*(q^6)*(r^(-15))))^(-4)
```

Nomor 3
Kalkulasikan!

$$\frac{4(8-6)^2 - 4 \cdot 3 + 2 \cdot 8}{3^1 + 19^0}$$

```
>$ (4*(8-6)^2-4*3+2*8)/3^1+19^0
```

Sederhanakan!
Nomor 4

$$(m^{x-b} \cdot n^{x+b})^x(m^bn^{-b})^x$$

```
>$ ((((m^(x-b))*n^(x+b))^x)*((m^b)*(n^(-b))^x))
```

Nomor 5

$$[\frac{(3x^a y^b)^3}{(-3x^a y^b)^2}]^2$$

```
>$ (((3*(x^a)*(y^b))^3)/(-3*(x^a)*(y^b))^2)^2
```

R.3

Lakukan operasi yang ditunjukkan!

Nomor 1

$$(3a^2)(-7a^4)$$

jawaban manual:

$$(3a^2)(-7a^4) = (3)(-7)(a^{2+4}) = -21a^6$$

jawaban emt:

```
>$ (3*a^2)*(-7*a^4)
```

Nomor 2

$$(2x+3y)^2$$

```
>$ (2*x+3*y)^2
```

Nomor 3

$$(x+1)(x-1)(x^2+1)$$

```
>$ (x+1)*(x-1)*(x^2+1)
```

Nomor 4

$$(z+4)(z-2)$$

```
>$ (z+4)*(z-2)
```

Nomor 5

$$(a^n+b^n)^2$$

```
>$ (a^n+b^n)^2
```

R.4

Faktorkan!
Nomor 1

$$t^2+8t+15$$

jawaban manual:

$$t^2+8t+15=(t+5)(t+3)$$

jawaban emt:

```
>$&factor(t^2+8*t+15)
```

Nomor 2

$$y^2 + 12y + 27$$

```
>$&factor(y^2+12*y+27)
```

Nomor 3

$$5m^4 - 20$$

```
>$&factor(5*(m^4)-20)
```

Nomor 4

$$6x^2 - 6$$

```
>$&factor(6*x^2-6)
```

Nomor 5

$$4t^3 + 108$$

```
>$&factor(4*t^3+108)
```

R.5

Hitung!
Nomor 1

$$7(3x + 6) = 11 - (x + 2)$$

jawaban manual:

$$7(3x+6)=11-(x+2)$$

$$21x+42=11-x-2$$

$$22x=33$$

$$x=\frac{-}{3}2$$

jawaban emt:

```
>$&solve(7*(3*x+6)=11-(x+2), x)
```

Nomor 2

$$x^2+5x$$

```
>$&solve(x^2+5*x, x)
```

Nomor 3

$$y^2+6y+9=0$$

```
>$&solve(y^2+6*y+9=0, y)
```

Nomor 4

$$n^2+4n+4=0$$

```
>$&solve(n^2+4*n+4=0, n)
```

Nomor 5

$$6x^2 - 7x = 10$$

```
>$&solve(6*x^2-7*x=10, x)
```

R.6

Kali atau bagi, dan sederhanakan!
Nomor 1

$$\frac{r-s}{r+s} \cdot \frac{r^2-s^2}{(r-s)^2}$$

```
>$&(((r-s)/(r+s))*((r^2-s^2)/(r-s)^2)); $&factor(%)
```

Nomor 2

$$\frac{x^2-4}{x^2-4x+4}$$

```
>$&((x^2-4)/(x^2-4*x+4)); $&factor(%)
```

Nomor 3

$$\frac{4-x}{x^2+4x-32}$$

```
>$&((4-x)/(x^2+4*x-32)); $&factor(%)
```

Nomor 4

$$\frac{7}{5x} + \frac{3}{5x}$$

jawaban manual:

$$\frac{7}{5x}+\frac{3}{5x}=\frac{7+3}{5x}=\frac{10}{5x}=\frac{2}{x}$$

jawaban emt:

```
>$&((7/(5*x))+(3/(5*x))); $&factor(%)
```

Nomor 5

$$\frac{5}{4z}-\frac{3}{8z}$$

```
>$&((5/4*z)-(3/8*z)); $&(%)
```

REVIEW EXERCISES

Nomor 1

$$(x^n+10)(x^n-4)$$

jawaban manual:

$$(x^n+10)(x^n-4)=x^{2n}-4x^n+10x^n-40=x^{2n}+6x^n-40$$

jawaban emt:

```
>$&expand((x^n+10)*(x^n-4))
```

Nomor 2

$$(t^a+t^{-a})^2$$

```
>$&expand((t^a+t^(-a))^2)
```

Nomor 3

$$(a^n - b^n)^3$$

```
>$&expand((a^n-b^n)^3)
```

Nomor 4

$$y^{2n} + 16y^n + 64$$

```
>$& factor(y^(2*n)+16*y^n+64)
```

Nomor 5

$$m^{6n} - m^{3n}$$

```
>$&factor((m^(6*n))-(m^(3*n)))
```

2.3 Exercise Set

Diberikan bahwa

$$f(x) = 3x + 1, g(x) = x^2 - 2x - 6, h(x) = x^3,$$

temukan masing-masing dari yang berikut ini.
Nomor 1

$$(f \circ g)(-1)$$

jawaban manual :

$$(f \circ g)(-1) = 3((-1)^2 - 2(-1) - 6) + 1 = 3(1 + 2 - 6) + 1 = 3(-3) + 1 = -8$$

jawaban emt:

```
>function f(x):= 3*x+1;
>function g(x):= x^2-2*x-6;
>function h(x):= x^3
>f(g(-1))
```

```
-8.00
```

Nomor 2

$$(h \circ f)(1)$$

```
>h(f(1))
```

```
64.00
```

Nomor 3

$$(f \circ h)(-3)$$

```
>f(h(-3))
```

```
-80.00
```

Nomor 4

$$(f \circ f)(-4)$$

```
>f(f(-4))
```

```
-32.00
```

Nomor 5

$$(f \circ g)(1/3)$$

```
>f(g(1/3))
```

```
-18.67
```

3.1 Exercise Set

Sederhanakan. Tulis jawaban dalam bentuk a+bi, di mana a dan b adalah bilangan real.
Nomor 1

$$(-5 + 3i) + (7 + 8i)$$

jawaban manual:

$$(-5 + 3i) + (7 + 8i) = -5 + 7 + 3i + 8i = 2 + 11i$$

jawaban emt:

```
>$ ((-5+3*i)+(7+8*i))
```

Nomor 2

$$(12 + 3i) + (-8 + 5i)$$

```
>$((12+3*i)+(-8+5*i))
```

Nomor 3

$$7i(2 - 5i)$$

```
>$&expand((7*i)*(2-5*i))
```

Nomor 4

$$-2i(-8+3i)$$

```
>$&expand(2*i*(-8+3*i))
```

Nomor 5

$$(10-4i)-(8+2i)$$

```
>$((10-4*i)-(8+2*i))
```

3.4 Exercise Set

Cari solusinya

Nomor 1

$$\frac{1}{4}+\frac{1}{5}=\frac{1}{t}$$

jawaban manual:

$$\frac{1}{4}+\frac{1}{5}=\frac{1}{t}$$

$$\frac{5+4}{20}=1t$$

$$9t=20$$

$$t=\frac{20}{9}$$

jawaban emt:

```
>$&solve((1/4)+(1/5)=(1/t),t)
```

Nomor 2

$$x + \frac{6}{x} = 5$$

```
>$&solve(x+(6/x)=5, x)
```

Nomor 3

$$\sqrt{3x-4} = 1$$

```
>$&solve(sqrt(3*x-4)=1, x)
```

Nomor 4

$$\sqrt(4)x^2 - 1 = 1$$

```
>$&solve(sqrt(4)*x^2-1=1, x)
```

Nomor 5

$$\sqrt{y+4} - \sqrt{y-1} = 1$$

```
>$&solve(sqrt(y+4)-sqrt(y-1)=1, y)
```

Hitung!
Nomor 1

$$|2x| \geq 6$$

```
>&load(fourier_elim)
```

```
        C:/Program Files/Euler x64/maxima/share/maxima/5.35.1/share/f\
  ourier_elim/fourier_elim.lisp
```

```
>$&fourier_elim(abs(2*x)>= 6, [x])
```

Nomor 2

$$|x+8| < 9$$

```
>$&fourier_elim(abs(x+8)< 9, [x])
```

Nomor 3

$$|x-5| > 0.1$$

```
>$&fourier_elim(abs(x-5)> 0.1, [x])
```

Nomor 4

$$|4x| > 20$$

```
>$&fourier_elim(abs(4*x)> 20, [x])
```

Nomor 5

$$|x+6|>10$$

```
>$&fourier_elim(abs(x+6)> 10, [x])
```

Chapter 3 Test

Cari solusinya
Nomor 1

$$x+5\sqrt{x}-36=0$$

```
>$&solve(x+5*sqrt(x)-36=0, x)
```

Nomor 2

$$\frac{3}{3x+4}+\frac{2}{x-1}=2$$

```
>$&solve((3/(3*x+4))+(2/(x-1))=2, x)
```

Nomor 3

$$\sqrt{x+4}-2=1$$

```
>$&solve(sqrt(x+4)-2=1, x)
```

Nomor 4

$$\sqrt{x+4}-\sqrt{x-4}=2$$

```
>$&solve(sqrt(x+4)-sqrt(x-4)=2, x)
```

4.1 Exercise Set

Nomor 1
Gunakan substitusi untuk menentukan apakah 4, 5, dan -2 adalah akar dari

$$f(x) = x^3 - 9x^2 + 14x + 24$$

jawaban manual:

$$4^3 - 9(4)^2 + 14(4) - 24 = 0$$

$$5^3 - 9(5)^2 + 14(5) - 24 \neq 0$$

$$(-2)^3 - 9(-2)^2 + 14(-2) - 24 \neq 0$$

Jadi, 4 adalah akar dari fungsi tersebut, sedangkan 5 dan -2 bukan.
jawaban emt:

```
>function f(x):=x^3-9*x^2+14*x+24
>f(4)
```

```
0.00
```

```
>f(5)
```

```
-6.00
```

```
>f(-2)
```

```
-48.00
```

Nomor 2
Gunakan substitusi untuk menentukan apakah 2, 3, dan -1 adalah akar dari

$$f(x) = 2x^3 - 3x^2 + x + 6.$$

```
>function f(x) :=2*x^3-3*x^2+x+6
>f(2)
```

```
        12.00
```

```
>f(3)
```

```
        36.00
```

```
>f(-1)
```

```
         0.00
```

Jadi, -1 adalah akar dari fungsi tersebut, sedangkan 2 dan 3 bukan.

Cari akar-akarnya
Nomor 3

$$f(x) = (x^2 - 5x + 6)^2$$

```
>$& solve((x^2-5*x+6)^2=0, x)
```

Nomor 4

$$f(x) = x^4 - 4x^2 + 3$$

```
>$& solve(x^4-4*x^2+3=0, x)
```

Nomor 5

$$f(x) = x^3 - x^2 - 8x + 4$$

```
>$& solve(x^3-x^2-8*x+4=0, x)
```

4.3 Exercise Set

Faktorkan fungsi polinomial, kemudian selesaikan persamaannya f(x)=0.
Nomor 1

$$f(x) = x^3 + 4x^2 + x - 6$$

```
>$&factor(x^3+4*x^2+x-6), $&solve(%)
```

Nomor 2

$$f(x) = x^3 + 2x^2 - 13x + 10$$

```
>$&factor(x^3+2*x^2-13*x+10), $&solve(%)
```

Nomor 3

$$x^3 - 3x^2 - 10x + 24$$

```
>$&factor(x^3-3*x^2-10*x+24), $&solve(%)
```

Nomor 4

$$x^4 - x^3 - 19x^2 + 49x - 30$$

```
>$&factor(x^4-x^3-19*x^2+49*x-30), $&solve(%)
```

Nomor 5

$$f(x) = x^4 + 11x^3 + 41x^2 + 61x + 30$$

```
>$&factor(x^4+11*x^3+41*x^2+61*x+30), $&solve(%)
```

BAB 2

MENGGAMBAR PLOT 2D DENGAN EMT

Menggambar Grafik 2D dengan EMT

Notebook ini menjelaskan tentang cara menggambar berbagaikurva dan grafik 2D dengan software EMT. EMT menyediakan fungsi plot2d() untuk menggambar berbagai kurva dan grafik dua dimensi (2D).

Plot Dasar

Ada fungsi plot yang sangat mendasar. Terdapat koordinat layar, yang selalu berkisar dari 0 hingga 1024 di setiap sumbu, tidak peduli apakah layarnya persegi atau tidak. Terdapat koordinat plot, yang dapat diatur dengan setplot(). Pemetaan antara koordinat tergantung pada jendela plot saat ini. Sebagai contoh, default shrinkwindow() menyisakan ruang untuk label sumbu dan judul plot.

Dalam contoh, kita hanya menggambar beberapa garis acak dalam berbagai warna. Untuk detail mengenai fungsi-fungsi ini, pelajari fungsi inti EMT.

```
>clg; // clear screen
>window(0,0,1024,1024); // use all of the window
>setplot(0,1,0,1); // set plot coordinates
>hold on; // start overwrite mode
>n=100; X=random(n,2); Y=random(n,2);  // get random points
>colors=rgb(random(n),random(n),random(n)); // get random colors
>loop 1 to n; color(colors[#]); plot(X[#],Y[#]); end; // plot
>hold off; // end overwrite mode
>insimg; // insert to notebook
```

```
>reset;
```

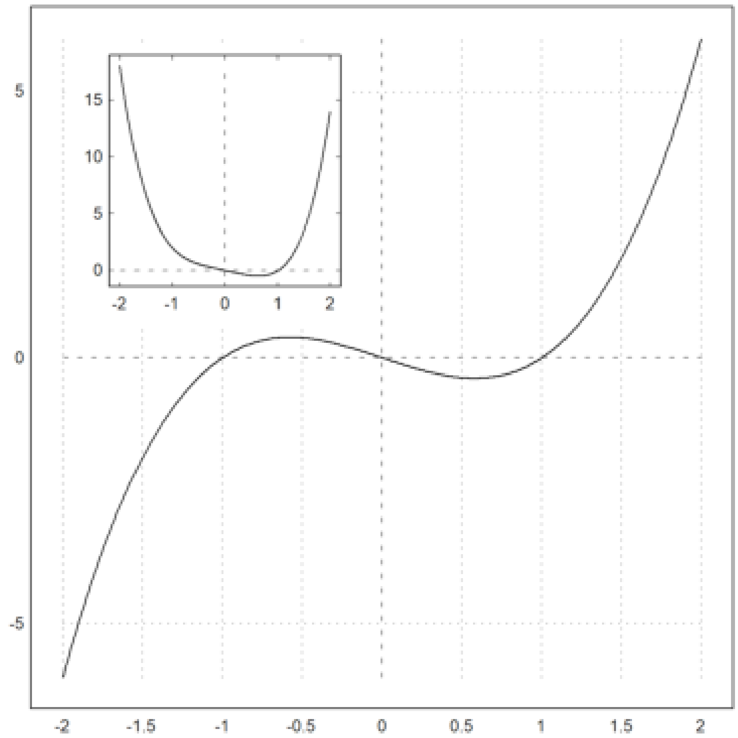
Anda harus menahan grafik, karena perintah plot() akan menghapus jendela plot

Untuk menghapus semua yang telah kita lakukan, kita menggunakan reset().

Untuk menampilkan gambar hasil plot di layar notebook, perintah plot2d() dapat diakhiri dengan titik dua (:). Cara lain adalah perintah plot2d() diakhiri dengan titik koma (;), kemudian menggunakan perintah insimg() untuk menampilkan gambar hasil plot.

Sebagai contoh lain, kita menggambar plot sebagai inset dalam plot lain. Hal ini dilakukan dengan mendefinisikan jendela plot yang lebih kecil. Perhatikan bahwa jendela ini tidak menyediakan ruang untuk label sumbu di luar jendela plot. Kita harus menambahkan beberapa margin untuk hal ini sesuai kebutuhan. Perhatikan bahwa kita menyimpan dan mengembalikan jendela penuh, dan menahan plot saat ini sementara kita membuat inset.

```
>plot2d("x^3-x");
>xw=200; yw=100; ww=300; hw=300;
>ow=window(); //window digunakan untuk mengatur jendela plot
>window(xw,yw,xw+ww,yw+hw);
>hold on;//mengaktifkan penahanan grafis.
>barclear(xw-50,yw-10,ww+60,ww+60);
>plot2d("x^4-x",grid=6):
```



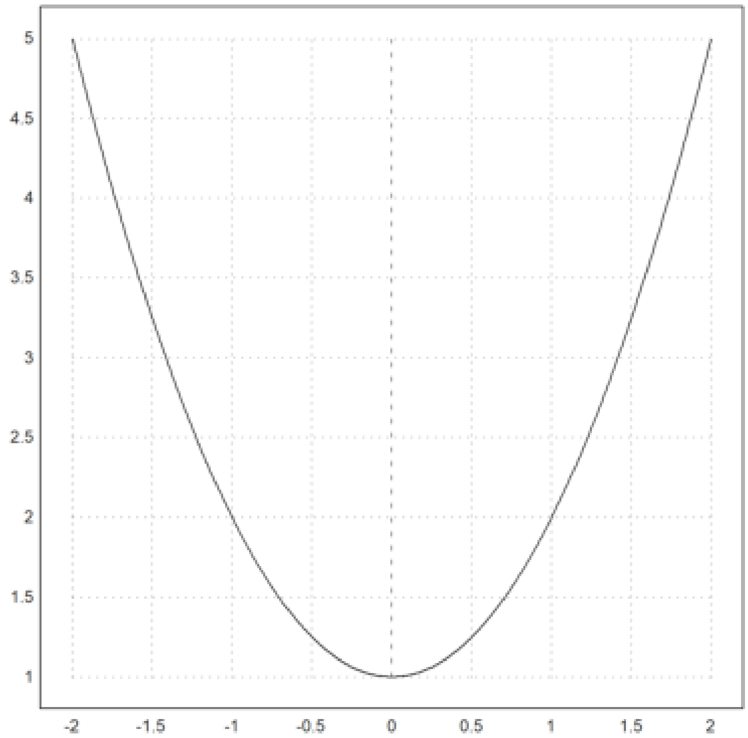

```
>  hold off;//menonaktifkan penahanan grafis.
>window(ow);
```

Plot dengan beberapa angka dicapai dengan cara yang sama. Ada fungsi utility figure() untuk ini.

Contoh tambahan :
Tentukan plot dari fungsi

$$f(x) : x^2 + 1$$

```
>plot2d("x^2+1",grid=2):
```



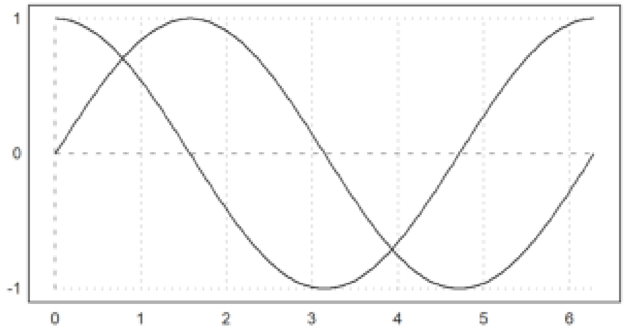

```
>reset;
```

Aspek Plot

Plot default menggunakan jendela plot persegi. Anda dapat mengubahnya dengan fungsi aspect(). Jangan lupa untuk mengatur ulang aspeknya nanti. Anda juga dapat mengubah default ini di menu dengan "Set Aspect" ke rasio aspek tertentu atau ke ukuran jendela grafis saat ini.

Tetapi Anda juga dapat mengubahnya untuk satu plot. Untuk melakukan ini, ukuran area plot saat ini diubah, dan jendela diatur sedemikian rupa sehingga label memiliki ruang yang cukup.

```
>aspect(2); // rasio panjang dan lebar 2:1
>plot2d(["sin(x)","cos(x)"],0,2pi):
```




```
>aspect();
>reset;
```

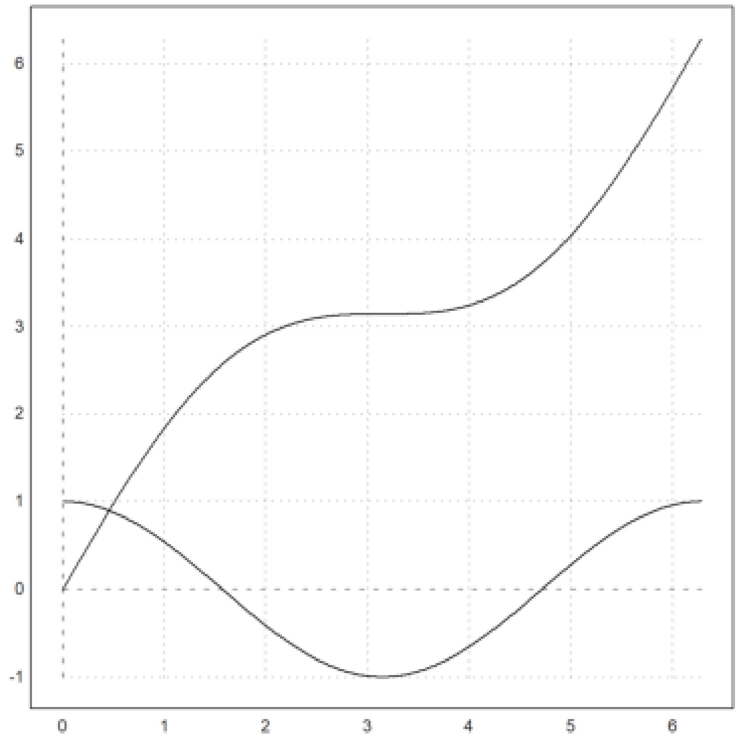
Fungsi reset() memulihkan default plot, termasuk rasio aspek.

Contoh Tambahan :
Tentukan plot dari

$$f(x) = sin(x) + x$$

$$f(x) = cos(x)$$

```
>aspect(1);
>plot2d(["sin(x)+x","cos(x)"],0,2pi):
```




```
>aspect();
>reset;
```

Plot 2D di Euler

EMT Math Toolbox memiliki plot dalam bentuk 2D, baik untuk data maupun fungsi. EMT menggunakan fungsi plot2d. Fungsi ini dapat memplot fungsi dan data.

Hal ini memungkinkan untuk memplot di Maxima menggunakan Gnuplot atau di Python menggunakan Math Plot Lib. Euler dapat memplot plot 2D dari
- ekspresi
- fungsi, variabel, atau kurva yang diparameterkan,
- vektor nilai x-y,
- awan titik-titik di dalam pesawat,
- kurva implisit dengan level atau wilayah level.
- Fungsi yang kompleks

Gaya plot mencakup berbagai gaya untuk garis dan titik, plot batang dan plot berbayang.

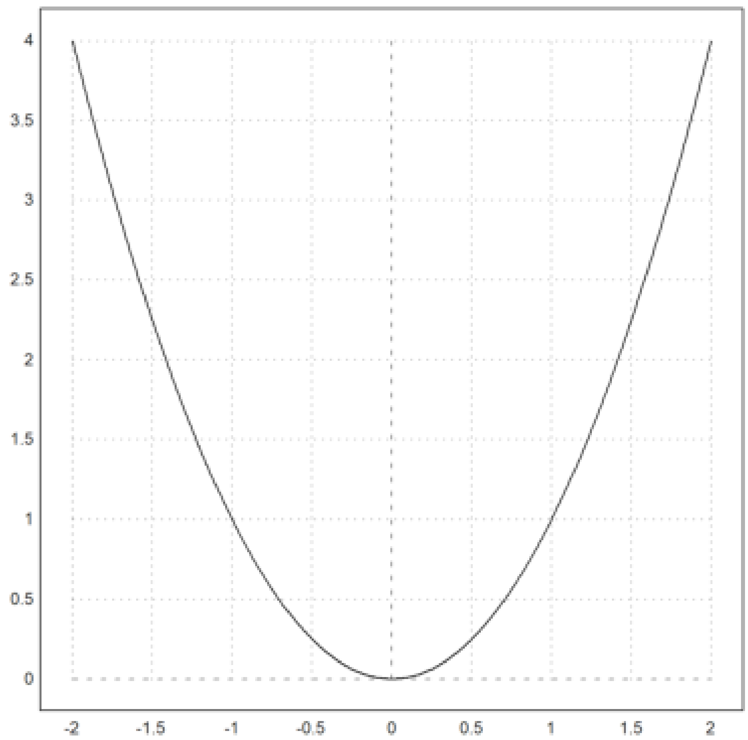
Plot Ekspresi atau Variabel

Sebuah ekspresi tunggal dalam "x" (misalnya "4*x^2") atau nama fungsi (misalnya "f") menghasilkan grafik fungsi.

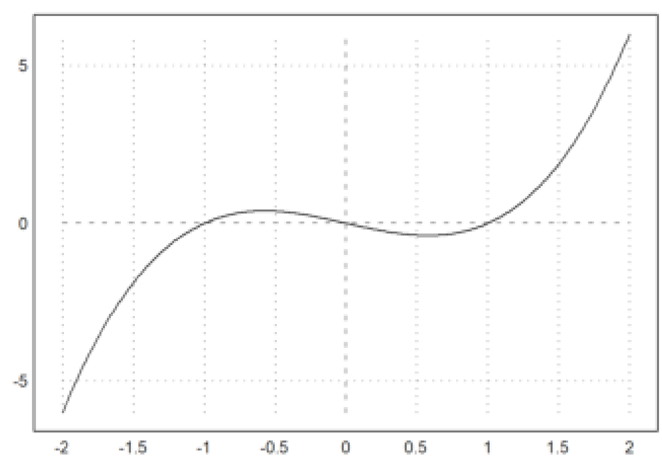
Berikut ini adalah contoh paling dasar, yang menggunakan rentang default dan menetapkan rentang y yang tepat agar sesuai dengan plot fungsi.

Catatan: Jika Anda mengakhiri baris perintah dengan tanda titik dua ":", plot akan disisipkan ke dalam jendela teks. Jika tidak, tekan TAB untuk melihat plot jika jendela plot tertutup.

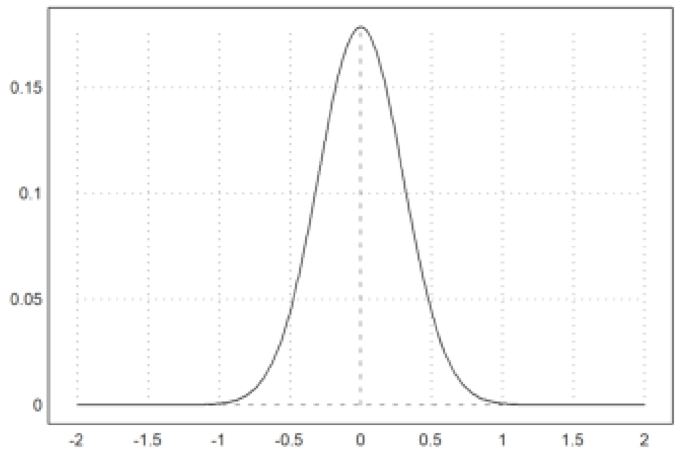
```
>plot2d("x^2"):
```




```
>aspect(1.5); plot2d("x^3-x"):
```



```
>a:=5.6; plot2d("exp(-a*x^2)/a"); insimg(30); // menampilkan gambar hasil p
```

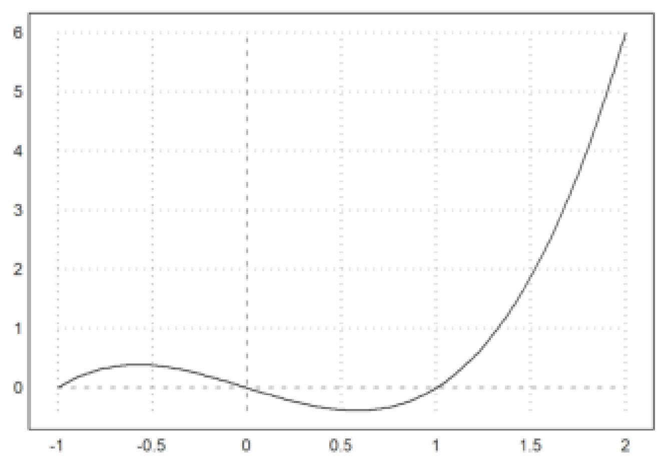


Dari beberapa contoh sebelumnya Anda dapat melihat bahwa aslinya gambar plot menggunakan sumbu X dengan rentang nilai dari -2 sampai dengan 2. Untuk mengubah rentang nilai X dan Y, Anda dapat menambahkan nilai-nilai batas X (dan Y) di belakang ekspresi yang digambar.

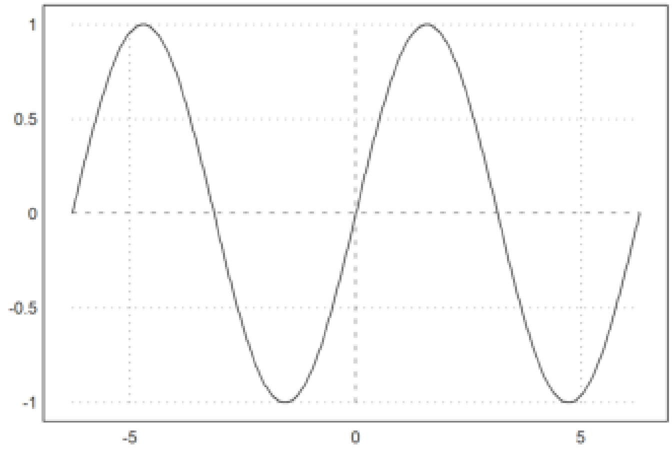
Rentang plot ditetapkan dengan parameter yang ditetapkan berikut ini

-a,b: rentang-x (default -2,2)
-c, d: rentang y (default: skala dengan nilai)
-r: sebagai alternatif adalah radius di sekitar pusat plot
-cx, cy: koordinat pusat plot (standar 0,0)

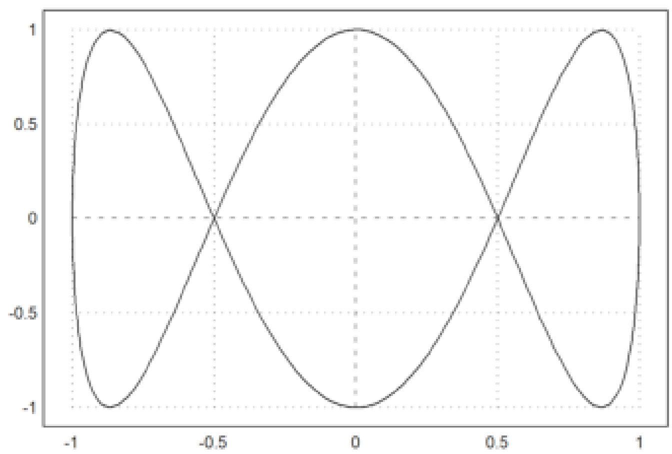
```
>plot2d("x^3-x",-1,2):
```



```
>plot2d("sin(x)",-2*pi,2*pi): // plot sin(x) pada interval [-2pi, 2pi]
```




```
>plot2d("cos(x)","sin(3*x)",xmin=0,xmax=2pi):
```



Alternatif untuk tanda titik dua adalah perintah insimg(lines), yang menyisipkan plot yang menempati sejumlah baris teks tertentu.

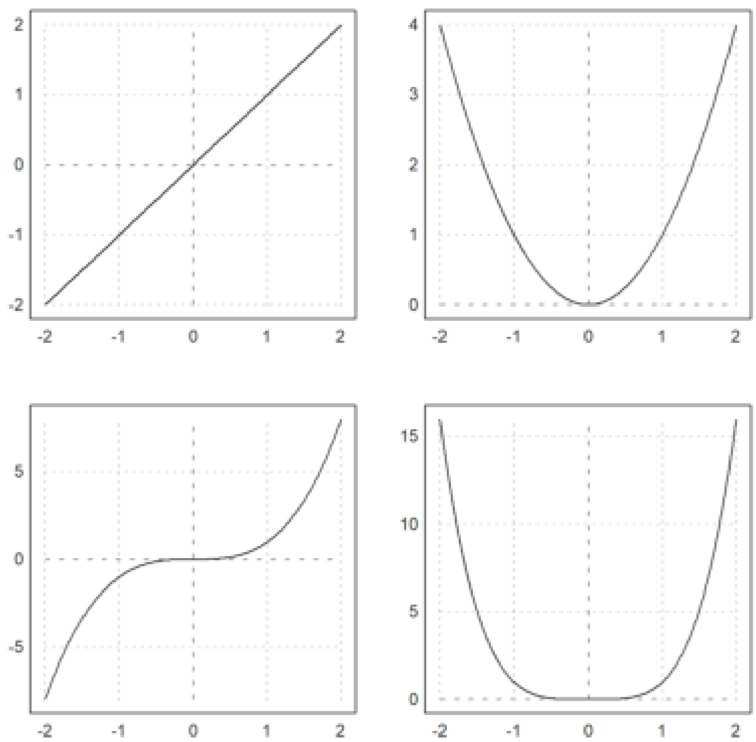
Dalam opsi, plot dapat diatur untuk muncul

- dalam jendela terpisah yang dapat diubah ukurannya,
- di jendela buku catatan.

Lebih banyak gaya dapat dicapai dengan perintah plot tertentu.

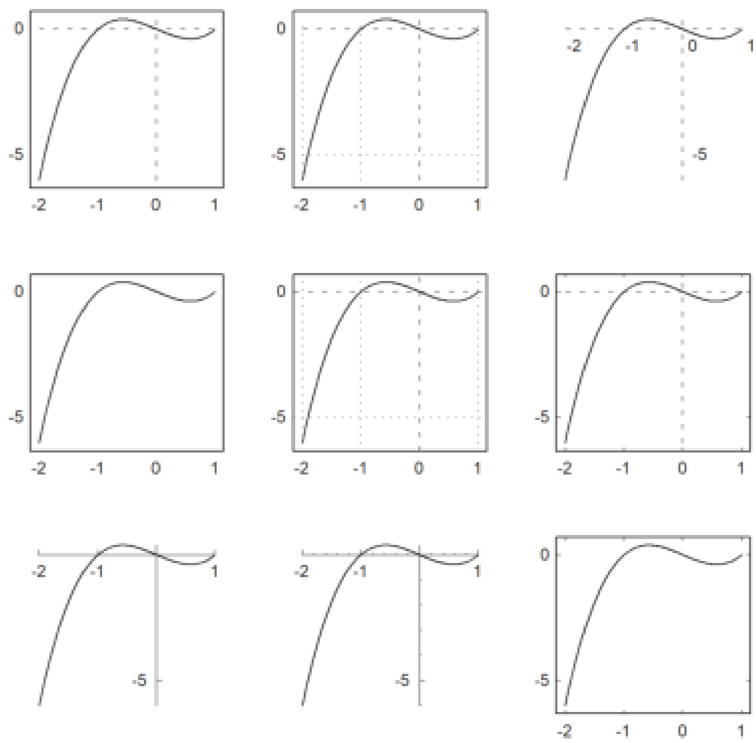
Dalam hal apa pun, tekan tombol tabulator untuk melihat plot, jika disembunyikan.
Untuk membagi jendela menjadi beberapa plot, gunakan perintah figure(). Pada contoh, kita memplot x^1 hingga x^4 menjadi 4 bagian jendela. figure(0) mengatur ulang jendela default.

```
>reset;
>figure(2,2); ...
>for n=1 to 4; figure(n); plot2d("x^"+n); end; ...
>figure(0):
```



Pada plot2d(), terdapat beberapa gaya alternatif yang tersedia dengan grid=x. Sebagai gambaran umum, kami menampilkan berbagai gaya grid dalam satu gambar (lihat di bawah ini untuk perintah figure()). Gaya grid=0 tidak disertakan. Gaya ini tidak menampilkan grid dan frame.

```
>figure(3,3); ...
>for k=1:9; figure(k); plot2d("x^3-x",-2,1,grid=k); end; ...
>figure(0):
```

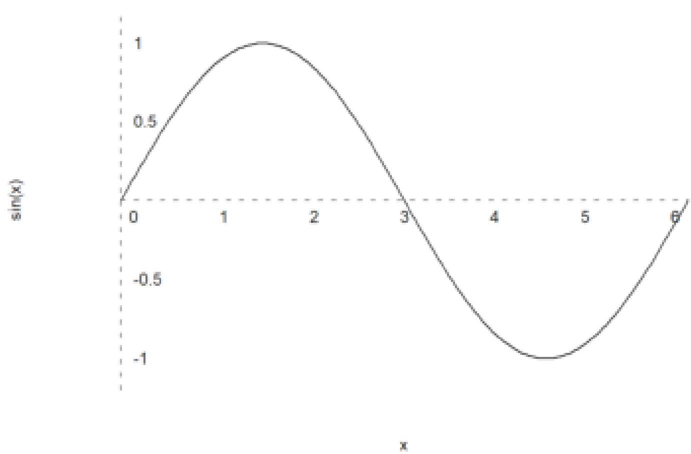


Jika argumen untuk plot2d() adalah sebuah ekspresi yang diikuti oleh empat angka, angka-angka ini adalah rentang x dan y untuk plot.

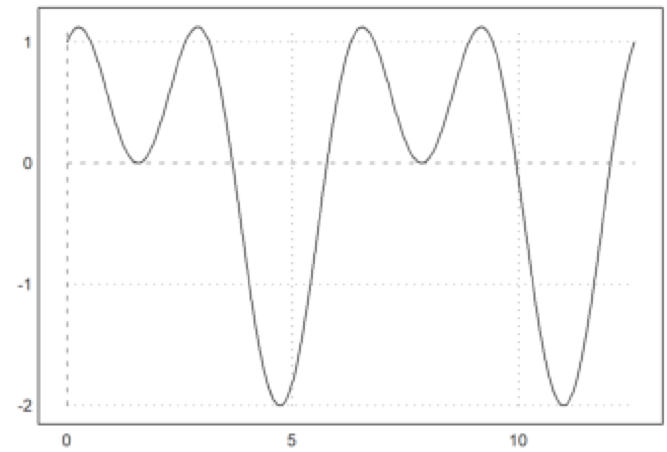
Atau, a, b, c, d dapat ditetapkan sebagai parameter yang ditetapkan sebagai a=... dst.

Pada contoh berikut, kita mengubah gaya kisi, menambahkan label, dan menggunakan label vertikal untuk sumbu y.

```
>aspect(1.5); plot2d("sin(x)",0,2pi,-1.2,1.2,grid=3,xl="x",yl="sin(x)"):
```



```
>plot2d("sin(x)+cos(2*x)",0,4pi):
```

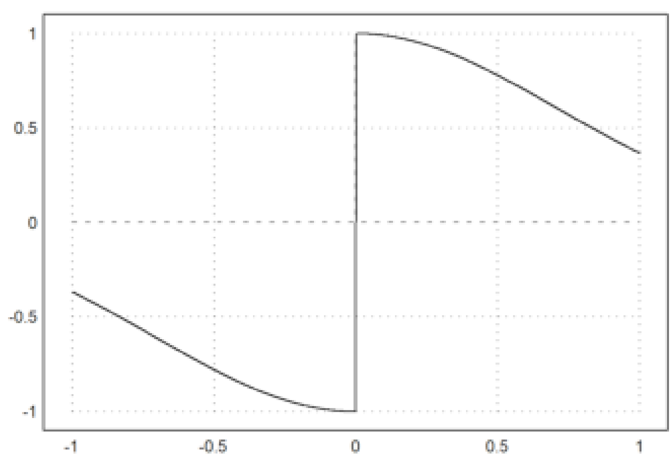


Gambar yang dihasilkan dengan menyisipkan plot ke dalam jendela teks disimpan di direktori yang sama dengan buku catatan, secara default dalam subdirektori bernama "images". Gambar-gambar tersebut juga digunakan oleh ekspor HTML.

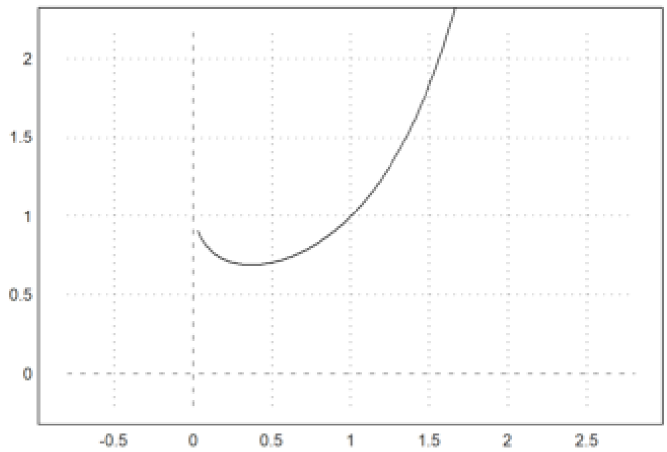
Anda bisa menandai gambar apa pun dan menyalinnya ke clipboard dengan Ctrl-C. Tentu saja, Anda juga dapat mengekspor grafik saat ini dengan fungsi-fungsi dalam menu File.

Fungsi atau ekspresi dalam plot2d dievaluasi secara adaptif. Untuk kecepatan yang lebih tinggi, matikan plot adaptif dengan <adaptive dan tentukan jumlah subinterval dengan n=... Hal ini hanya diperlukan pada kasus- kasus yang jarang terjadi.

```
>plot2d("sign(x)*exp(-x^2)",-1,1,<adaptive,n=10000):
```

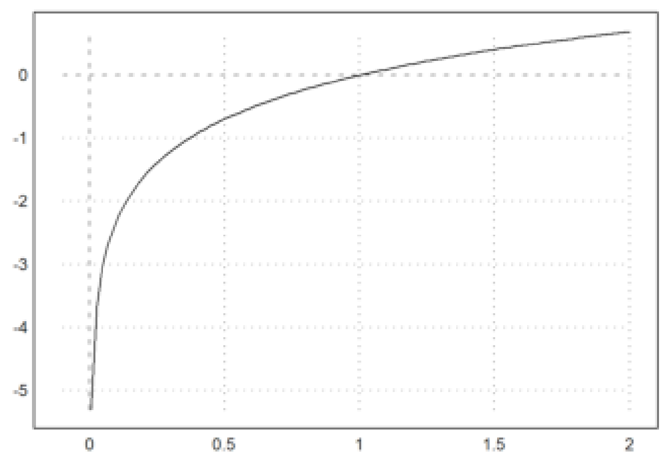


```
>plot2d("x^x",r=1.2,cx=1,cy=1):
```



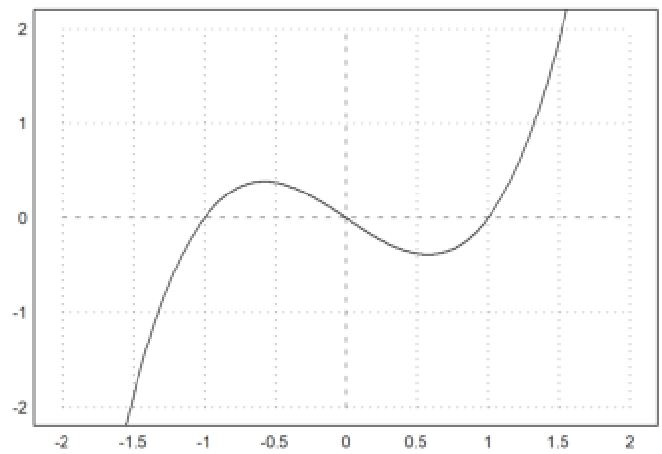
Perhatikan bahwa x^x tidak didefinisikan untuk x<=0. Fungsi plot2d menangkap kesalahan ini, dan mulai memplot segera setelah fungsi didefinisikan. Hal ini berlaku untuk semua fungsi yang mengembalikan NAN di luar jangkauan definisinya.

```
>plot2d("log(x)",-0.1,2):
```

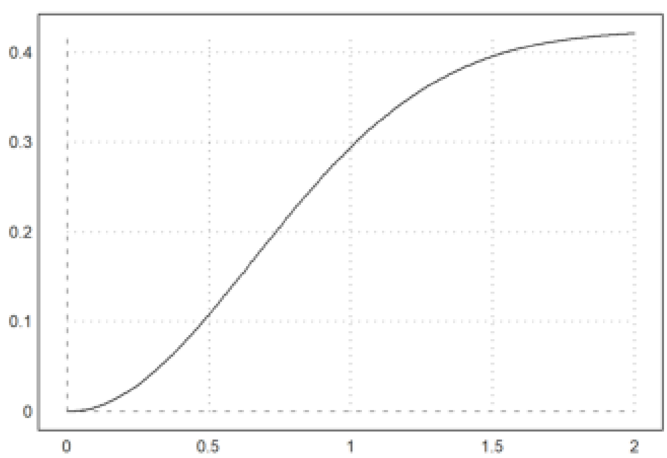


Parameter square=true (atau >square) memilih rentang y secara otomatis sehingga hasilnya adalah jendela plot persegi. Perhatikan bahwa secara default, Euler menggunakan ruang persegi di dalam jendela plot.

```
>plot2d("x^3-x",>square):
```

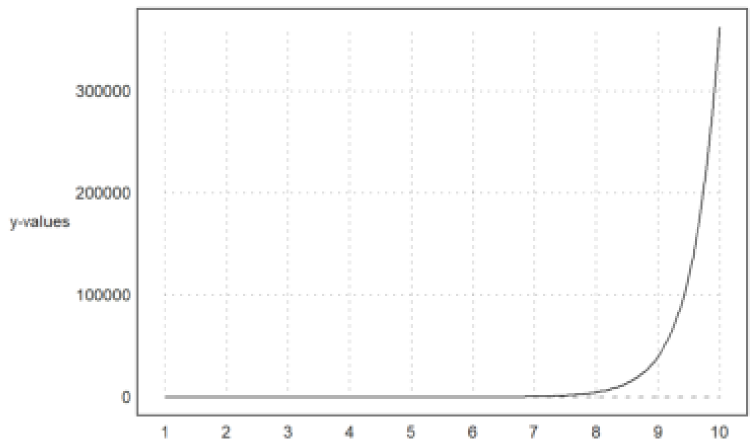



```
>plot2d(''integrate("sin(x)*exp(-x^2)",0,x)'',0,2): // plot integral
```



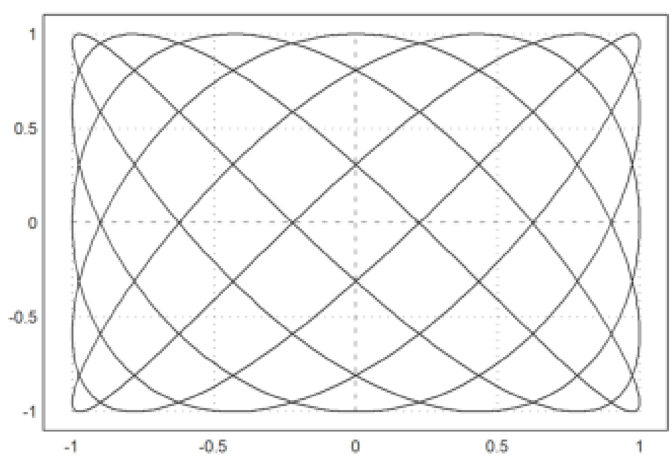
Jika Anda membutuhkan lebih banyak ruang untuk label-y, panggil shrinkwindow() dengan parameter lebih kecil, atau tetapkan nilai positif untuk "lebih kecil" pada plot2d().

```
>plot2d("gamma(x)",1,10,yl="y-values",smaller=6,<vertical):
```

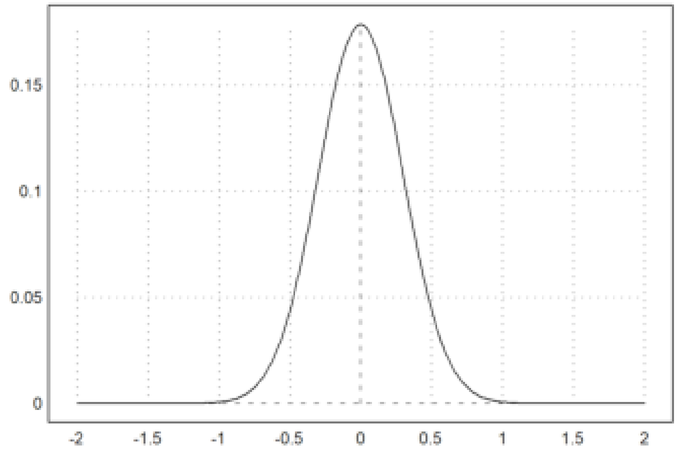


Ekspresi simbolik juga dapat digunakan, karena disimpan sebagai ekspresi string sederhana.

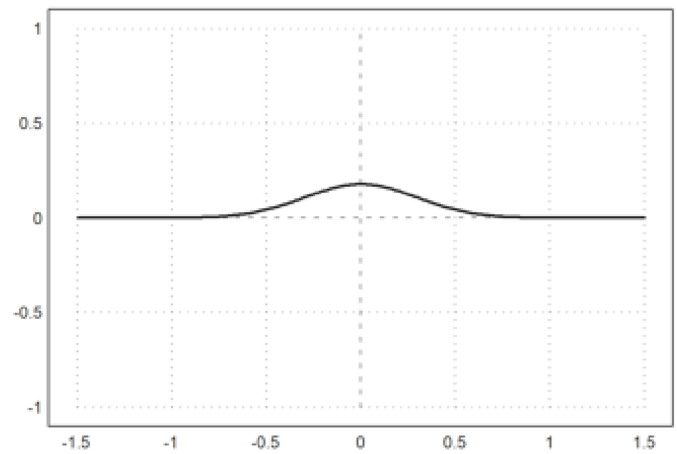
```
>x=linspace(0,2pi,1000); plot2d(sin(5x),cos(7x)):
```



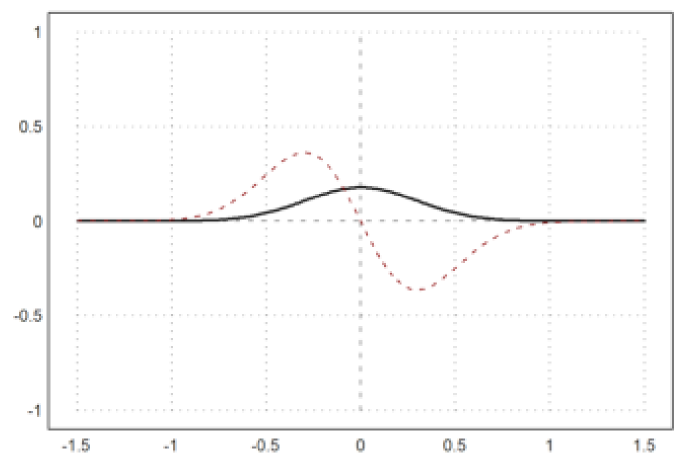

```
>a:=5.6; expr &= exp(-a*x^2)/a; // define expression
>plot2d(expr,-2,2): // plot from -2 to 2
```



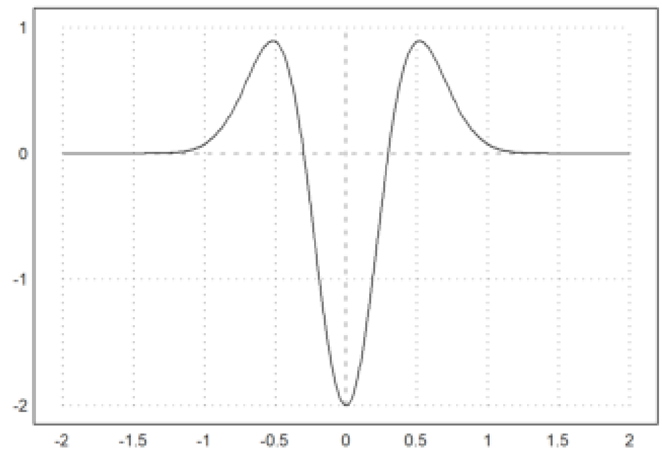

```
>plot2d(expr,r=1,thickness=2): // plot in a square around (0,0)
```



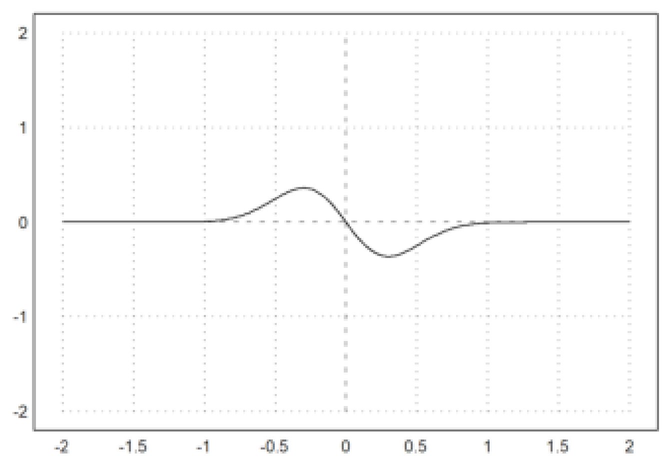

```
>plot2d(&diff(expr,x),>add,style="--",color=red): // add another plot
```



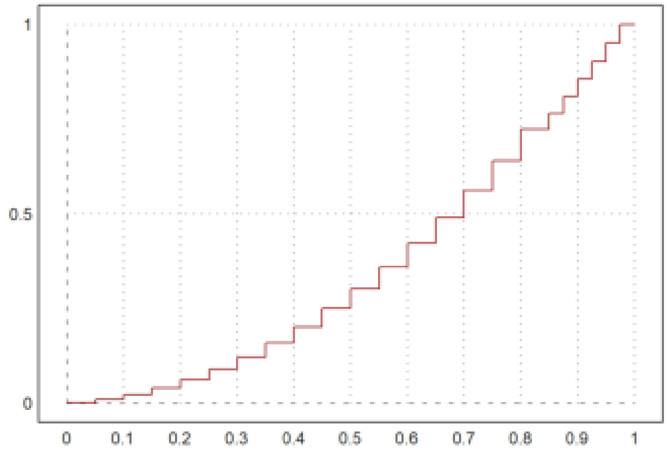
```
>plot2d(&diff(expr,x,2),a=-2,b=2,c=-2,d=1): // plot in rectangle
```



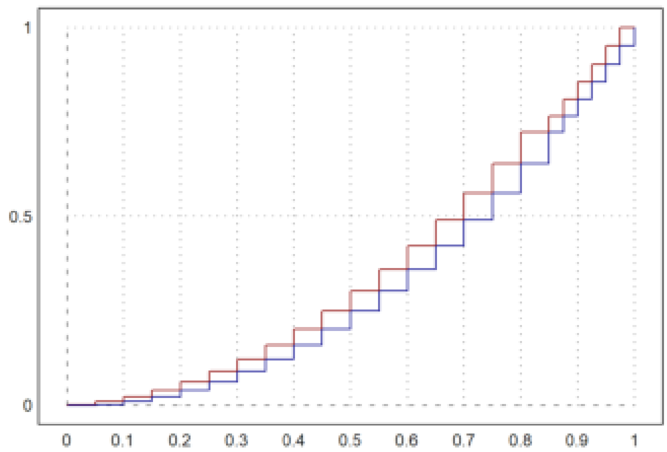

```
>plot2d(&diff(expr,x),a=-2,b=2,>square): // keep plot square
```




```
>plot2d("x^2",0,1,steps=1,color=red,n=10):
```




```
>plot2d("x^2",>add,steps=2,color=blue,n=10):
```

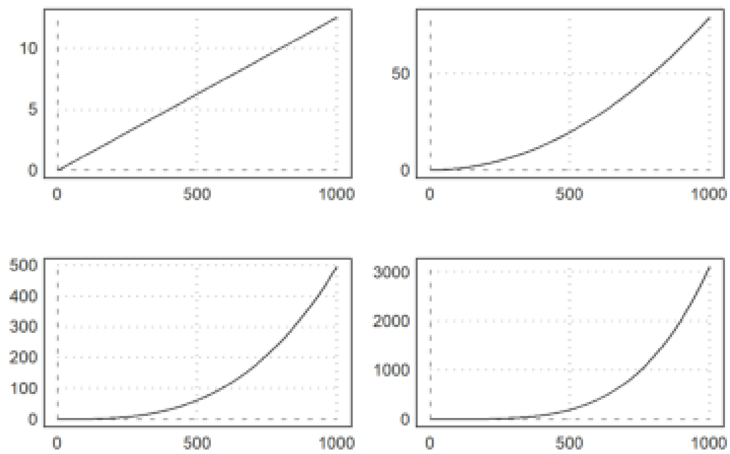


Contoh Tambahan :
1. Buat plot fungsi f(x) tersebut

$$f(x) = 2x^n, 1 \leq n \leq 4, n \in Z$$

penyelesaian :

```
>figure(2,2); ...
>for n=1 to 4; figure(n); plot2d(2*x^n); end; figure(0):
```

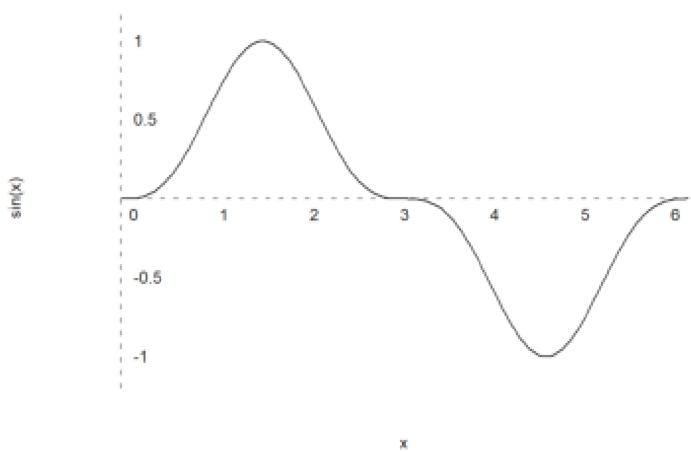


2. Buat plot fungsi berikut

$$f(x) = sin^3(x)$$

penyelesaian :

```
>aspect(1.5); plot2d("(sin(x))^3",0,2pi,-1.2,1.2,grid=3,xl="x",yl="sin(x)")
```

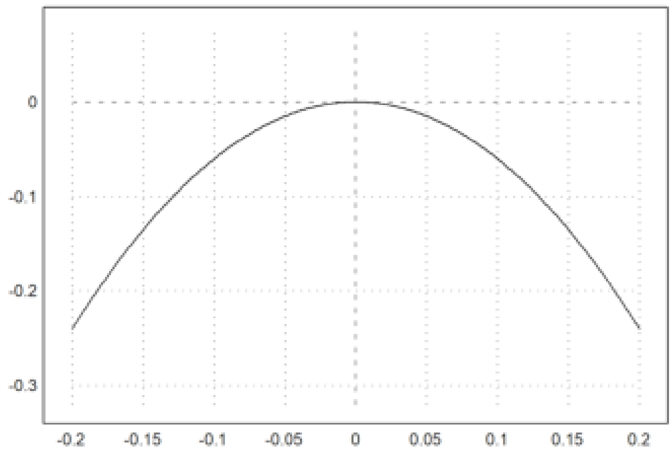


3. Buat plot turunan fungsi tersebut

$$f(x) = -2x^3$$

penyelesaian :

```
>plot2d(&diff(-2*x^3,x),a=-0.2,b=0.2,>square):
```

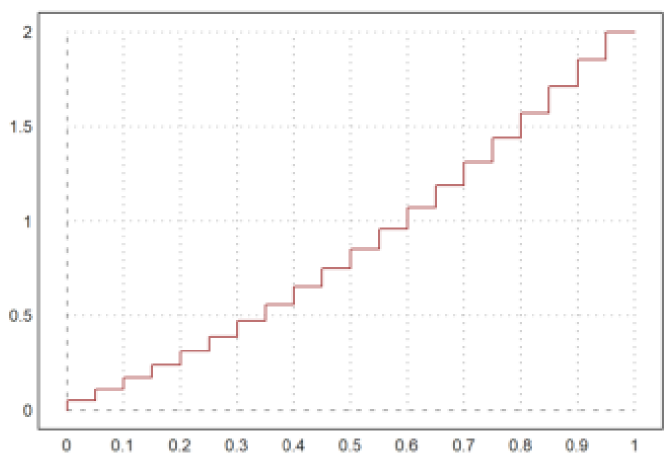


4. Buat plot fungsi tersebut sebagai fungsi tangga!

$$f(x) = x^2 + x$$

penyelesaian :

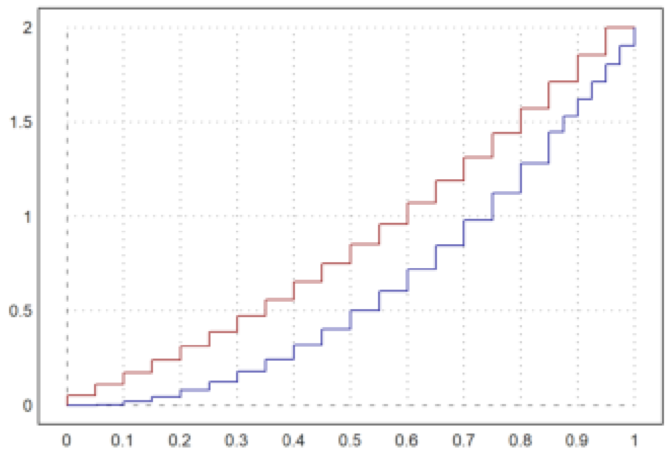
```
>plot2d("x^2+x",0,1,steps=1,color=red,n=10):
```



5. Tambahkan plot dari fungsi dibawah ke plot fungsi soal no 4!

$$f(x) = 2x^2$$

```
>plot2d("2*x^2",>add,steps=2,color=blue,n=10):
```

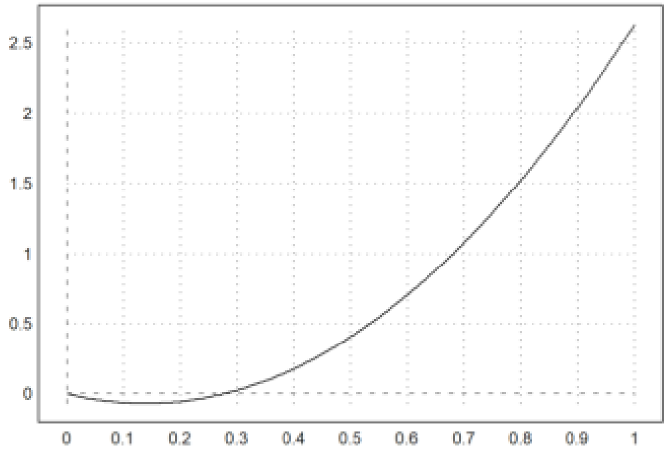


Fungsi dalam Satu Parameter

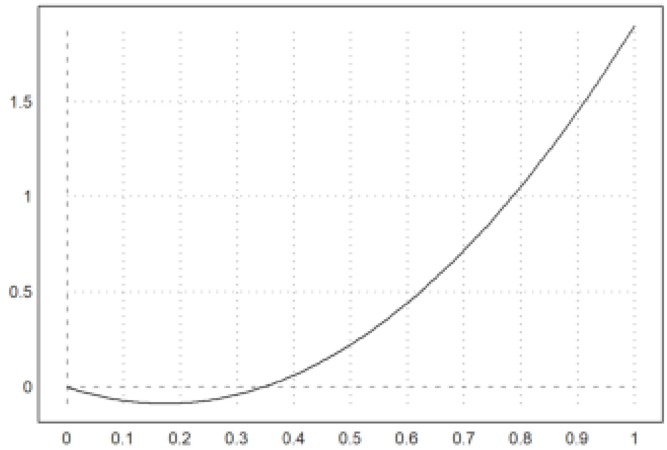
Fungsi plot yang paling penting untuk plot planar adalah plot2d(). Fungsi ini diimplementasikan dalam bahasa Euler dalam file "plot.e", yang dimuat pada awal program.

Berikut adalah beberapa contoh penggunaan fungsi. Seperti biasa dalam EMT, fungsi yang bekerja untuk fungsi atau eksekusi lain, Anda dapat mengoper parameter tambahan (selain x) yang bukan variabel global ke fungsi dengan parameter titik koma atau dengan koleksi panggilan.

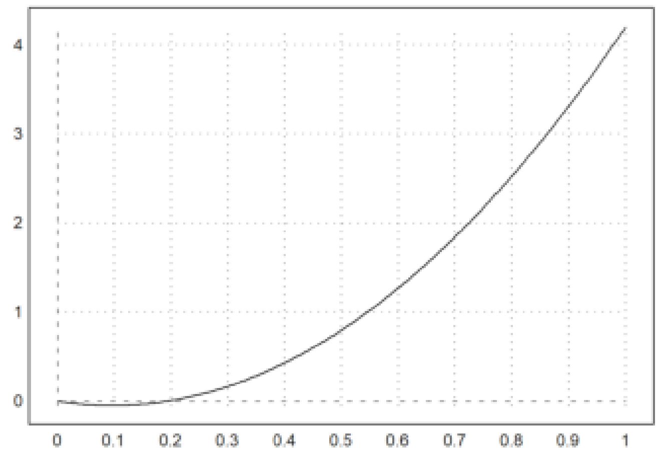
```
>function f(x,a) := x^2/a+a*x^2-x; // define a function
>a=0.3; plot2d("f",0,1;a): // plot with a=0.3
```



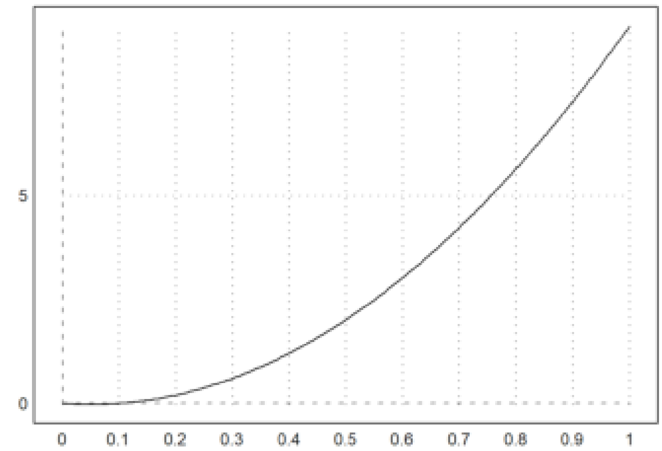

```
>plot2d("f",0,1;0.4): // plot with a=0.4
```



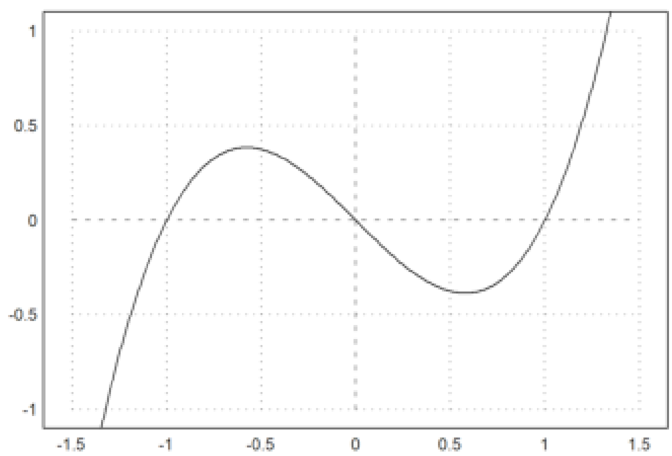

```
>plot2d({{"f",0.2}},0,1): // plot with a=0.2
```



```
>plot2d({{"f(x,b)",b=0.1}},0,1): // plot with 0.1
```




```
>function f(x) := x^3-x; ...
>plot2d("f",r=1):
```



Berikut ini adalah ringkasan dari fungsi yang diterima

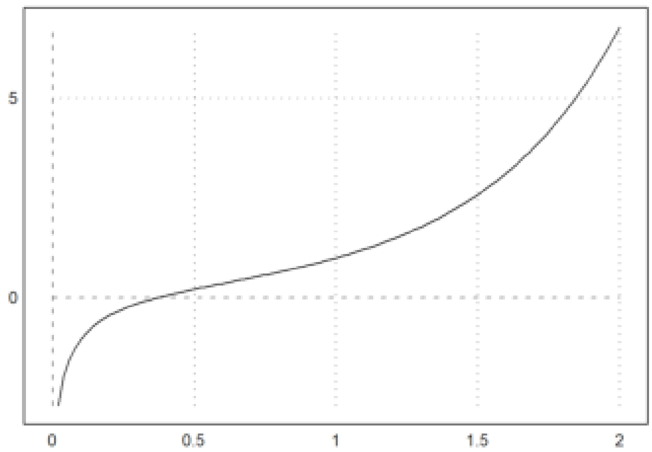
- ekspresi atau ekspresi simbolik dalam x
- fungsi atau fungsi simbolis dengan nama sebagai "f"
- fungsi simbolik hanya dengan nama f

Fungsi plot2d() juga menerima fungsi simbolik. Untuk fungsi simbolik, nama saja sudah cukup.

```
>function f(x) &= diff(x^x,x)
```

```
                    x
                   x  (log(x) + 1)
```

```
>plot2d(f,0,2):
```

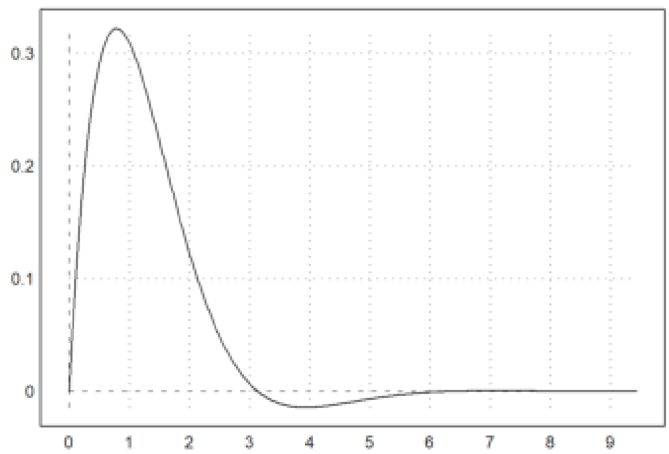


Tentu saja, untuk ekspresi atau ungkapan simbolik, nama variabel sudah cukup untuk memplotnya.

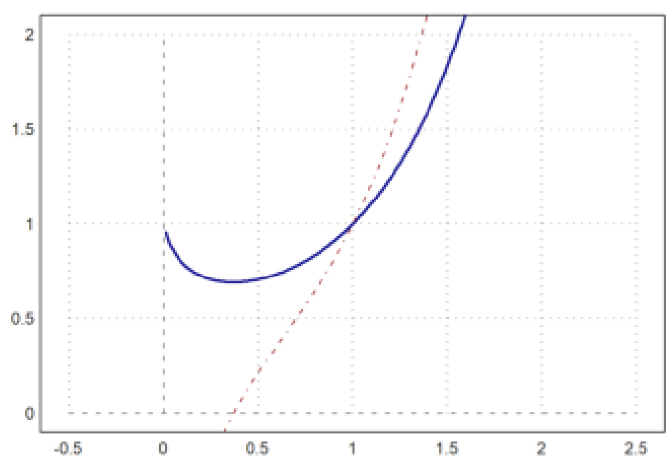
```
>expr &= sin(x)*exp(-x)
```

```
                    - x
                   E    sin(x)
```

```
>plot2d(expr,0,3pi):
```




```
>function f(x) &= x^x;
>plot2d(f,r=1,cx=1,cy=1,color=blue,thickness=2);
>plot2d(&diff(f(x),x),>add,color=red,style="-.-"):
```



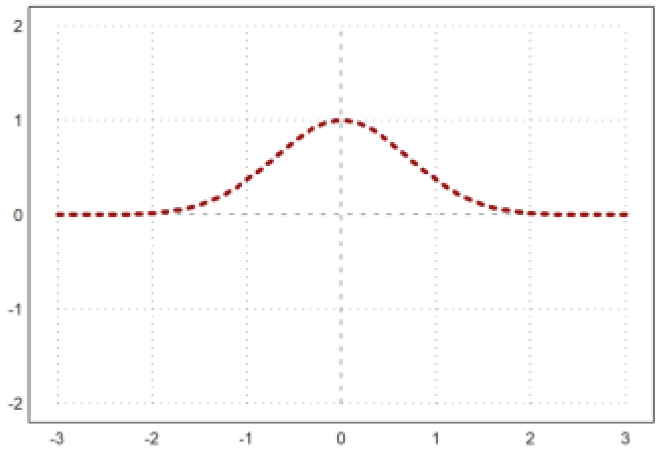
Untuk gaya garis, terdapat berbagai opsi.

- style = "...". Pilih dari "-", "–", "-.", ".", ".-.", "-.-".
- warna: Lihat di bawah untuk warna.
- ketebalan: Standarnya adalah 1.

Warna dapat dipilih sebagai salah satu warna default, atau sebagai warna RGB.

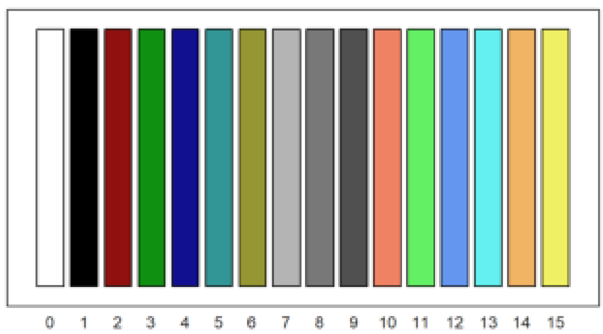
- 0..15: indeks warna default.
- konstanta warna: putih, hitam, merah, hijau, biru, cyan, zaitun, abu-abu muda, abu-abu, abu-abu tua, oranye, hijau muda, biru kehijauan, biru muda, oranye muda, kuning
- rgb (merah, hijau, biru): parameter dalam bentuk real dalam [0,1].

```
>plot2d("exp(-x^2)",r=2,color=red,thickness=3,style="--"):
```



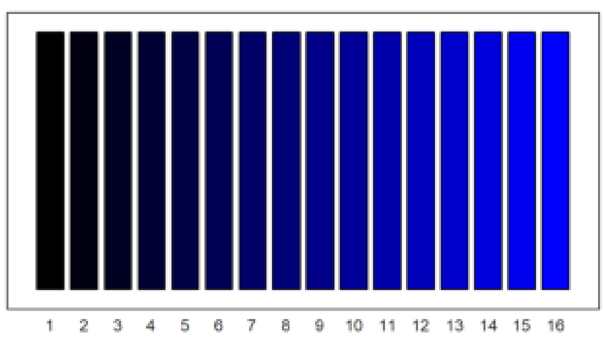
Berikut ini adalah pemandangan warna EMT yang sudah ditetapkan sebelumnya.

```
>aspect(2); columnsplot(ones(1,16),lab=0:15,grid=0,color=0:15):
```



Tetapi Anda bisa menggunakan warna apa pun.

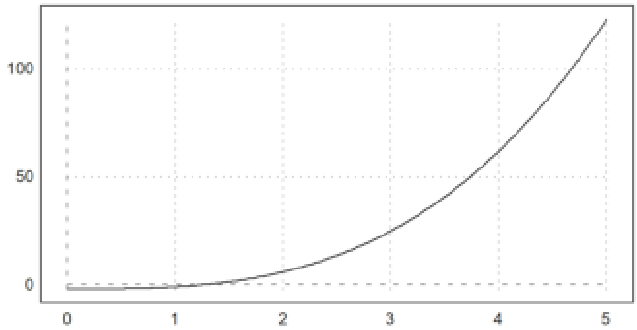
```
>columnsplot(ones(1,16),grid=0,color=rgb(0,0,linspace(0,1,15))):
```



Contoh Tambahan :
1. Buat plot dari fungsi tersebut!

$$f(x) = x^3 - 2$$

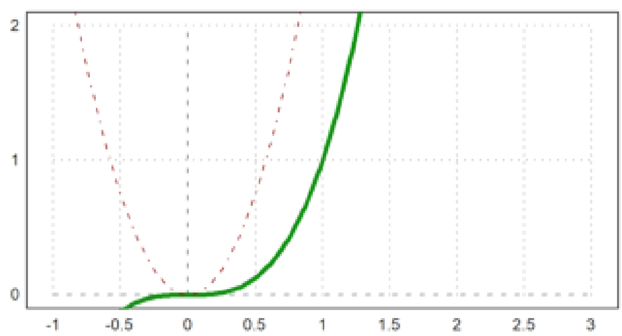
```
>function f(x)&=x^3-2;
>plot2d(f,0,5):
```



2. Buat plot dari fungsi tersebut!

$$f(x) = x^3$$

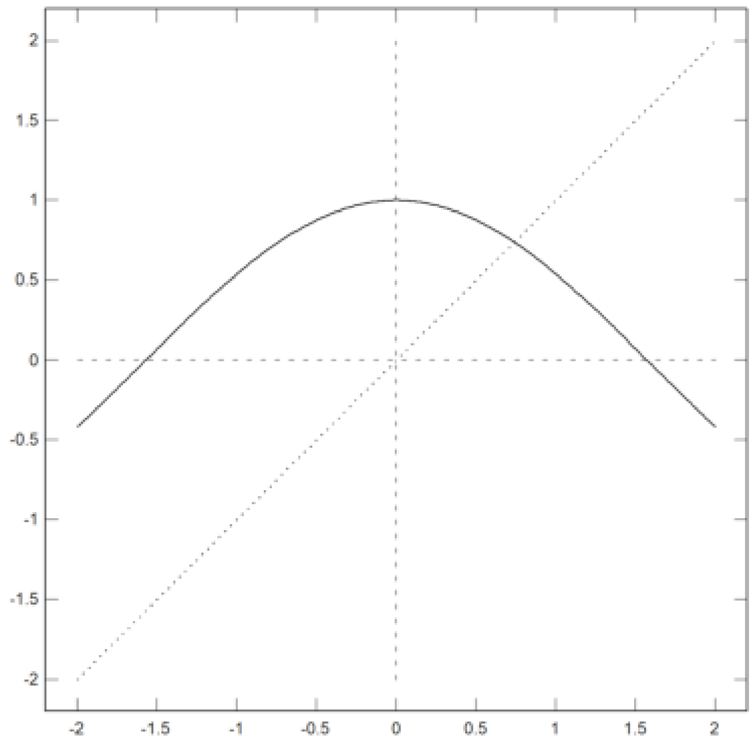
```
>function f(x) &= x^3;
>plot2d(f, r=1, cx=1, cy=1, color=green, thickness=3);
>plot2d(&diff(f(x),x),>add,color=red,style="-.-"):
```



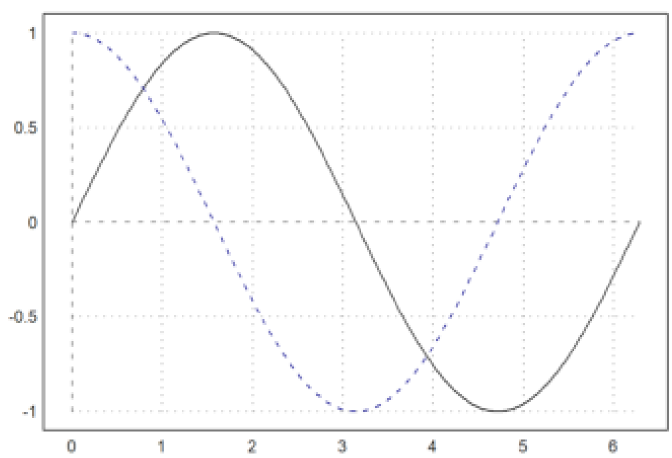
Menggambar Beberapa Kurva pada bidang koordinat yang sama

Memplot lebih dari satu fungsi (beberapa fungsi) ke dalam satu jendela dapat dilakukan dengan berbagai cara. Salah satu caranya adalah dengan menggunakan >add untuk beberapa pemanggilan ke plot2d secara bersamaan, kecuali pemanggilan pertama. Kita telah menggunakan fitur ini pada contoh di atas.

```
>aspect(); plot2d("cos(x)",r=2,grid=6); plot2d("x",style=".",>add):
```

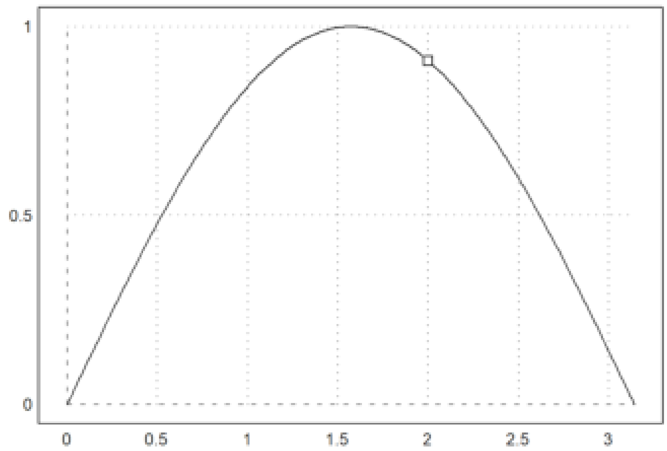



```
>aspect(1.5); plot2d("sin(x)",0,2pi); plot2d("cos(x)",color=blue,style="--"
```



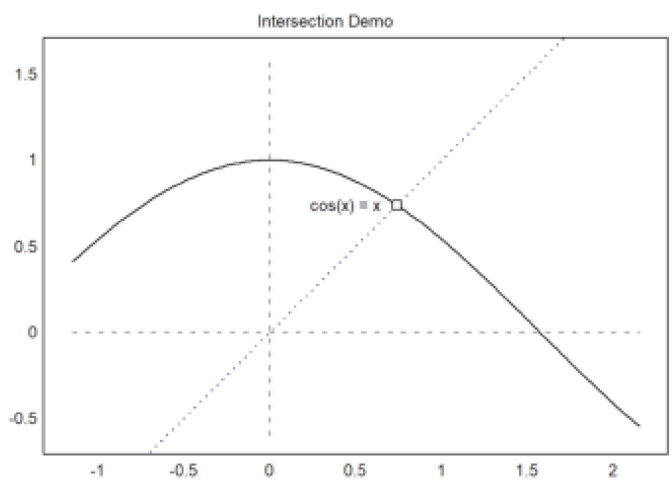
Salah satu kegunaan >add adalah untuk menambahkan titik pada kurva.

```
>plot2d("sin(x)",0,pi); plot2d(2,sin(2),>points,>add):
```



Kami menambahkan titik perpotongan dengan label (pada posisi "cl" untuk kiri tengah), dan menyisipkan hasilnya ke dalam buku catatan. Kami juga menambahkan judul ke plot.

```
>plot2d(["cos(x)","x"],r=1.1,cx=0.5,cy=0.5, ...
>  color=[black,blue],style=["-","."], ...
>  grid=1);
>x0=solve("cos(x)-x",1);  ...
>  plot2d(x0,x0,>points,>add,title="Intersection Demo");  ...
>  label("cos(x) = x",x0,x0,pos="cl",offset=20):
```



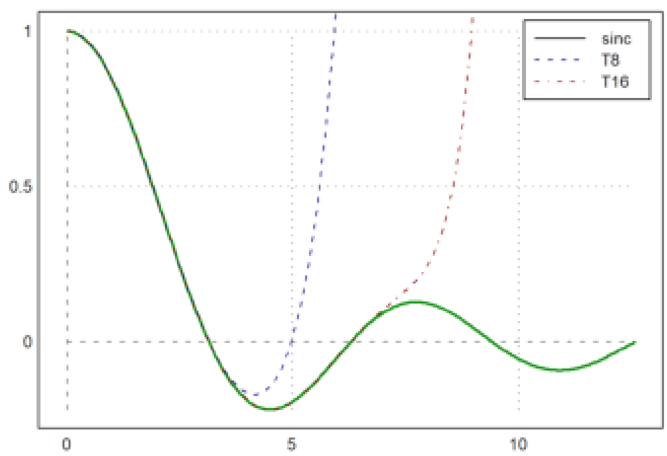
Dalam demo berikut ini, kami memplot fungsi sinc(x)=sin(x)/x dan ekspansi Taylor ke-8 dan ke-16. Kami menghitung ekspansi ini menggunakan Maxima melalui ekspresi simbolik.
Plot ini dilakukan dalam perintah multi-baris berikut ini dengan tiga kali pemanggilan plot2d(). Pemanggilan kedua dan ketiga memiliki set flag >add, yang membuat plot menggunakan rentang sebelumnya.

Kami menambahkan kotak label yang menjelaskan fungsinya.

```
>$taylor(sin(x)/x,x,0,4)
```

$$\frac{x^4}{120} - \frac{x^2}{6} + 1$$

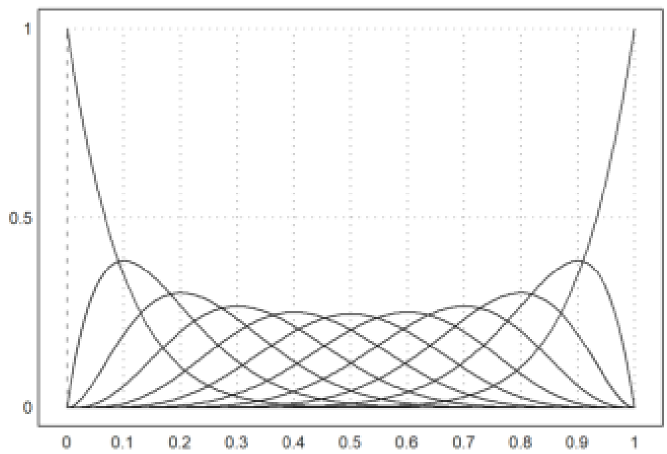
```
>plot2d("sinc(x)",0,4pi,color=green,thickness=2); ...
>  plot2d(&taylor(sin(x)/x,x,0,8),>add,color=blue,style="--"); ...
>  plot2d(&taylor(sin(x)/x,x,0,16),>add,color=red,style="-.-"); ...
>  labelbox(["sinc","T8","T16"],styles=["-","--","-.-"], ...
>    colors=[black,blue,red]):
```



Pada contoh berikut, kami menghasilkan Polinomial Bernstein.

$$B_i(x) = \binom{n}{i} x^i (1-x)^{n-i}$$

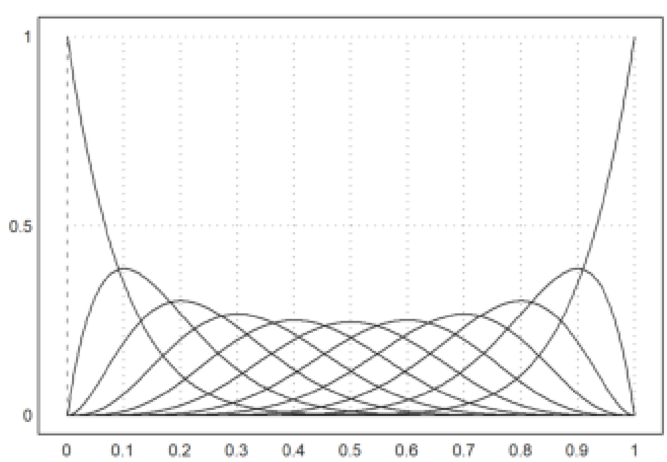
```
>plot2d("(1-x)^10",0,1); // plot first function
>for i=1 to 10; plot2d("bin(10,i)*x^i*(1-x)^(10-i)",>add); end;
>insimg;
```



Metode kedua adalah menggunakan sepasang matriks nilai x dan matriks nilai y dengan ukuran yang sama.

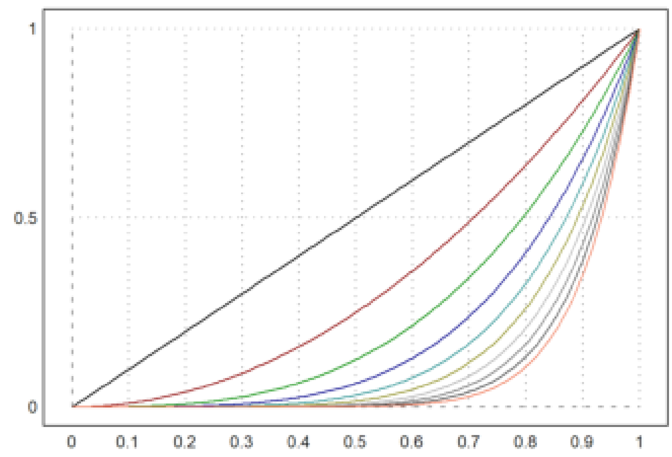
Kita membuat sebuah matriks nilai dengan satu Bernstein-Polynomial di setiap baris. Untuk ini, kita cukup menggunakan vektor kolom i. Lihatlah pengantar tentang bahasa matriks untuk mempelajari lebih lanjut.

```
>x=linspace(0,1,500);
>n=10; k=(0:n)'; // n is row vector, k is column vector
>y=bin(n,k)*x^k*(1-x)^(n-k); // y is a matrix then
>plot2d(x,y):
```



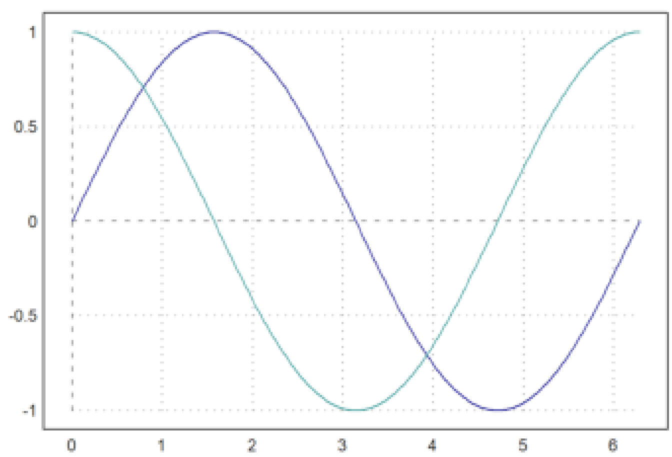
Perhatikan bahwa parameter warna dapat berupa vektor. Kemudian setiap warna digunakan untuk setiap baris matriks.

```
>x=linspace(0,1,200); y=x^(1:10)'; plot2d(x,y,color=1:10):
```

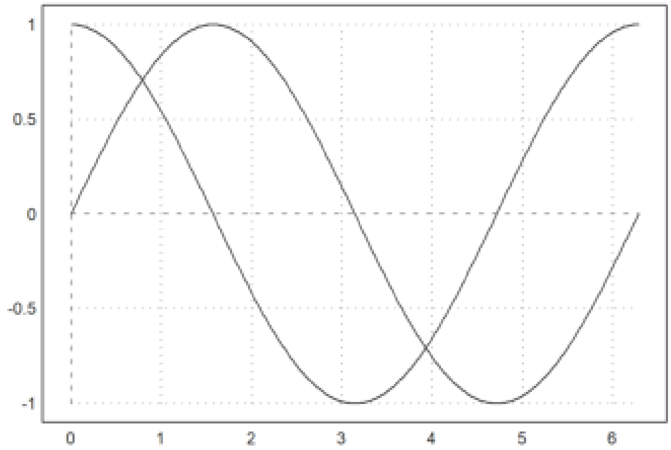


Metode lainnya adalah menggunakan vektor ekspresi (string). Anda kemudian dapat menggunakan larik warna, larik gaya, dan larik ketebalan dengan panjang yang sama

```
>plot2d(["sin(x)","cos(x)"],0,2pi,color=4:5):
```



```
>plot2d(["sin(x)","cos(x)"],0,2pi): // plot vector of expressions
```



Kita bisa mendapatkan vektor seperti itu dari Maxima dengan menggunakan makelist() dan mxm2str().

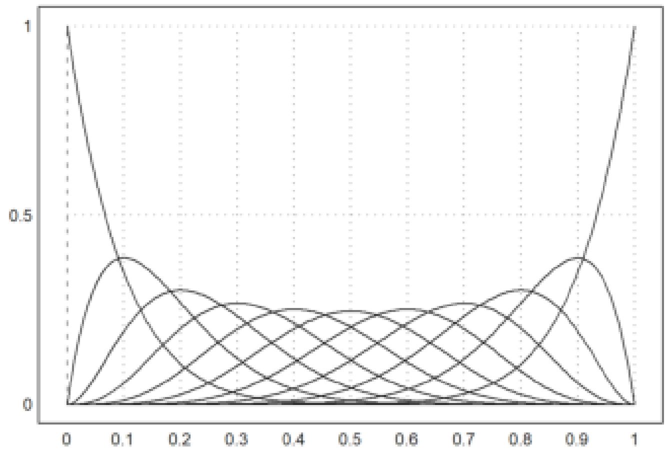
```
>v &= makelist(binomial(10,i)*x^i*(1-x)^(10-i),i,0,10) // make list
```

```
                 10               9                 8  2                 7  3
        [(1 - x)   , 10 (1 - x)   x, 45 (1 - x)   x , 120 (1 - x)   x ,
              6  4                5  5                4  6                3  7
 210 (1 - x)   x , 252 (1 - x)   x , 210 (1 - x)   x , 120 (1 - x)   x ,
             2  8               9   10
 45 (1 - x)   x , 10 (1 - x) x , x   ]
```

```
>mxm2str(v) // get a vector of strings from the symbolic vector
```

```
 (1-x)^10
 10*(1-x)^9*x
 45*(1-x)^8*x^2
 120*(1-x)^7*x^3
 210*(1-x)^6*x^4
 252*(1-x)^5*x^5
 210*(1-x)^4*x^6
 120*(1-x)^3*x^7
 45*(1-x)^2*x^8
 10*(1-x)*x^9
 x^10
```

```
>plot2d(mxm2str(v),0,1): // plot functions
```

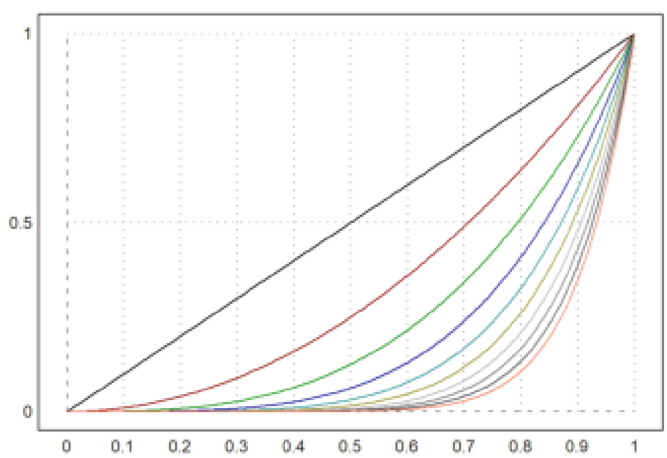


Alternatif lain adalah dengan menggunakan bahasa matriks Euler.

Jika sebuah ekspresi menghasilkan matriks fungsi, dengan satu fungsi di setiap baris, semua fungsi ini akan diplot ke dalam satu plot.

Untuk ini, gunakan vektor parameter dalam bentuk vektor kolom. Jika sebuah larik warna ditambahkan, maka akan digunakan untuk setiap baris plot.

```
>n=(1:10)'; plot2d("x^n",0,1,color=1:10):
```

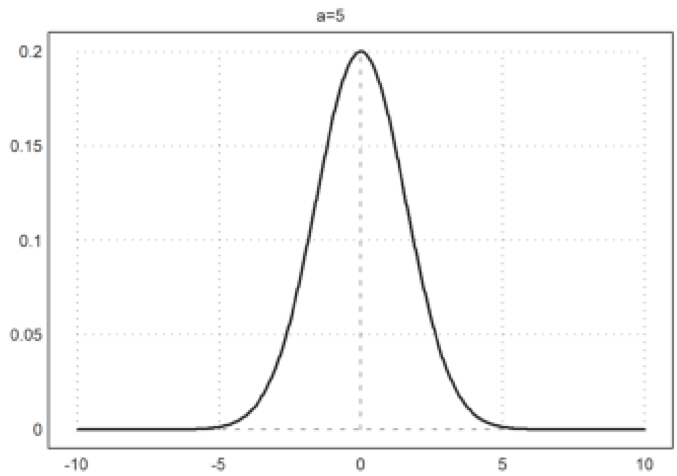


Ekspresi dan fungsi satu baris dapat melihat variabel global.

Jika Anda tidak dapat menggunakan variabel global, Anda perlu menggunakan fungsi dengan parameter tambahan, dan mengoper parameter ini sebagai parameter titik koma.

Berhati-hatilah untuk meletakkan semua parameter yang ditetapkan di akhir perintah plot2d. Pada contoh, kita memberikan a=5 ke fungsi f, yang kita plot dari -10 ke 10.

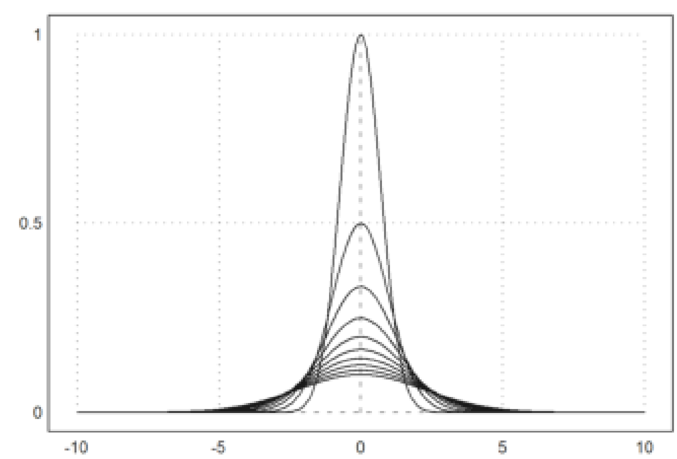
```
>function f(x,a) := 1/a*exp(-x^2/a); ...
>plot2d("f",-10,10;5,thickness=2,title="a=5"):
```



Atau, gunakan koleksi dengan nama fungsi dan semua parameter tambahan. Daftar khusus ini disebut koleksi panggilan, dan itu adalah cara yang lebih disukai untuk meneruskan argumen ke fungsi yang dengan sendirinya diteruskan sebagai argumen ke fungsi lain.

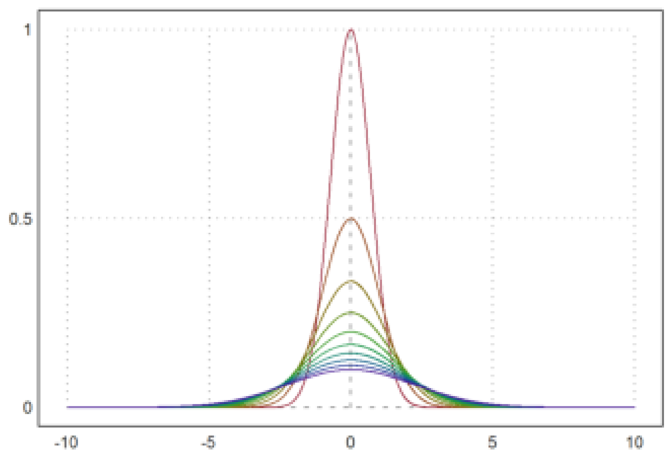
Pada contoh berikut ini, kita menggunakan loop untuk memplot beberapa fungsi (lihat tutorial tentang pemrograman untuk loop).

```
>plot2d({{"f",1}},-10,10); ...
>for a=2:10; plot2d({{"f",a}},>add); end:
```



Kita dapat mencapai hasil yang sama dengan cara berikut menggunakan bahasa matriks EMT. Setiap baris dari matriks f(x,a) adalah satu fungsi. Selain itu, kita dapat mengatur warna untuk setiap baris matriks. Klik dua kali pada fungsi getspectral() untuk penjelasannya.

```
>x=-10:0.01:10; a=(1:10)'; plot2d(x,f(x,a),color=getspectral(a/10)):
```

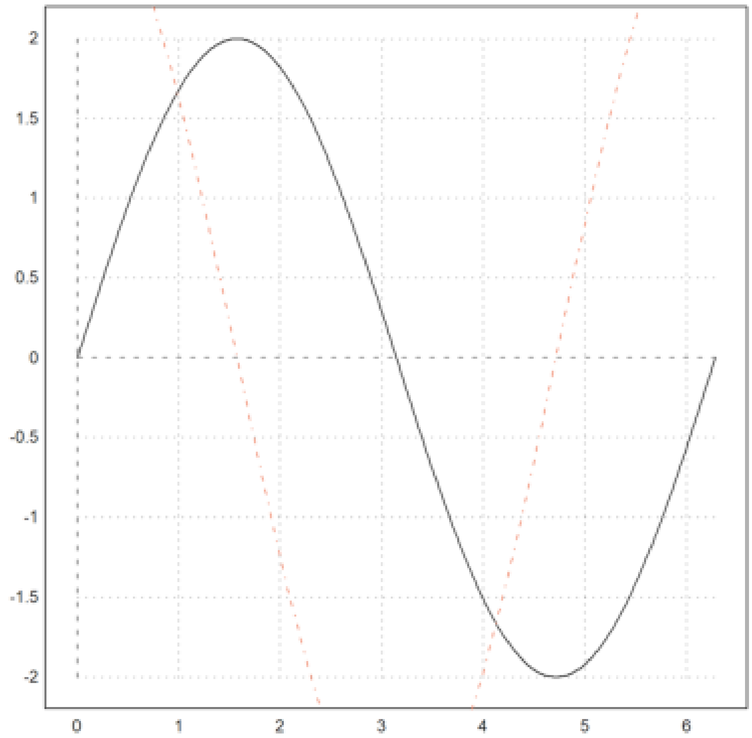


Contoh Tambahan :
1. Gabungkan plot dari dua fungsi tersebut

$$f(x) = 2sin(x)$$

$$f(x) = 3cos(x)$$

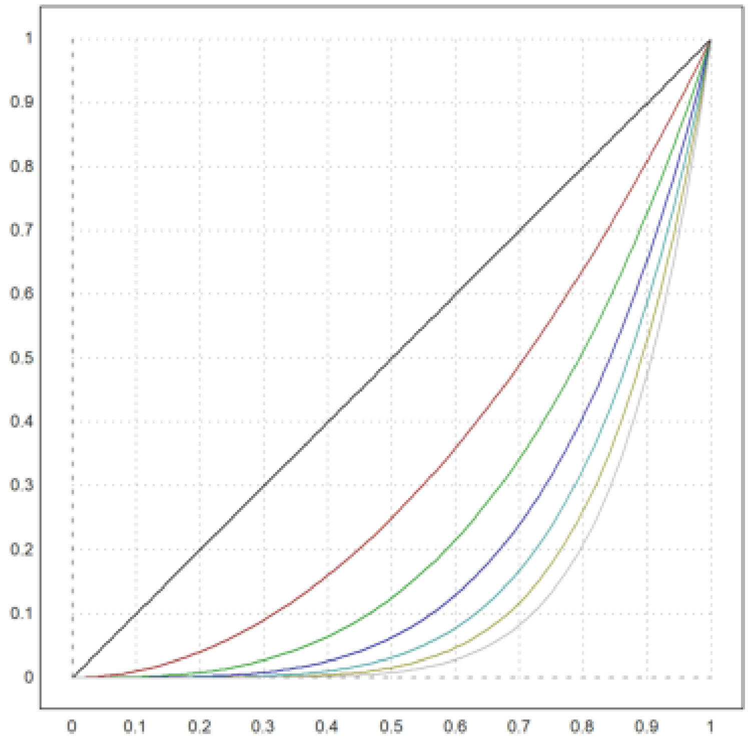
```
>aspect(1); plot2d("2*(sin(x))",0,2pi); plot2d("3*(cos(x))",color=orange,st
```



2. Buatlah plot dari fungsi berikut.

$$y = x^n, 1 \leq n \leq 7, n \in Z$$

```
>x=linspace(0,1,100); y=x^(1:7)'; plot2d(x,y,color=1:10):
```



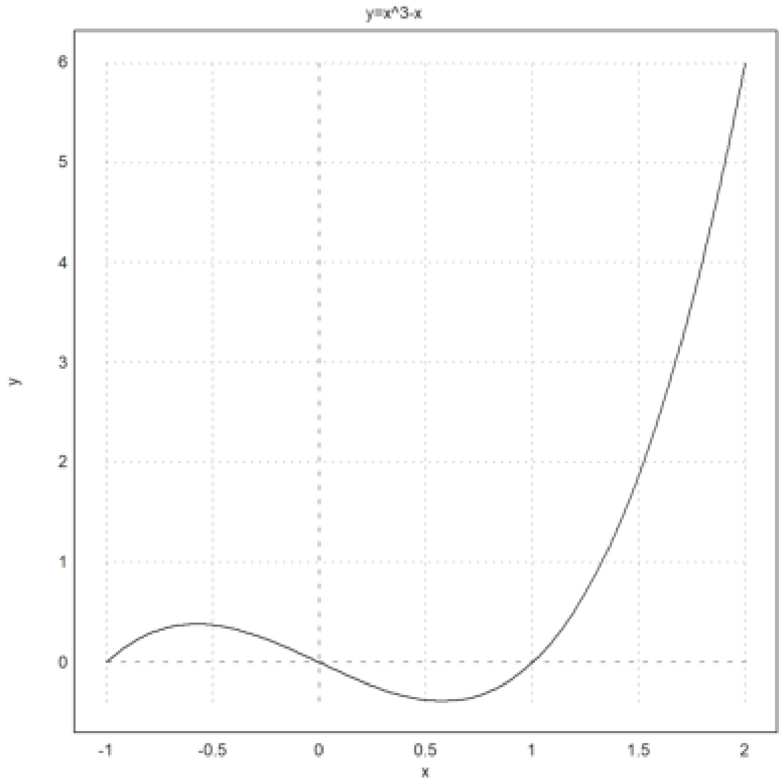
Label Teks

Dekorasi sederhana dapat berupa

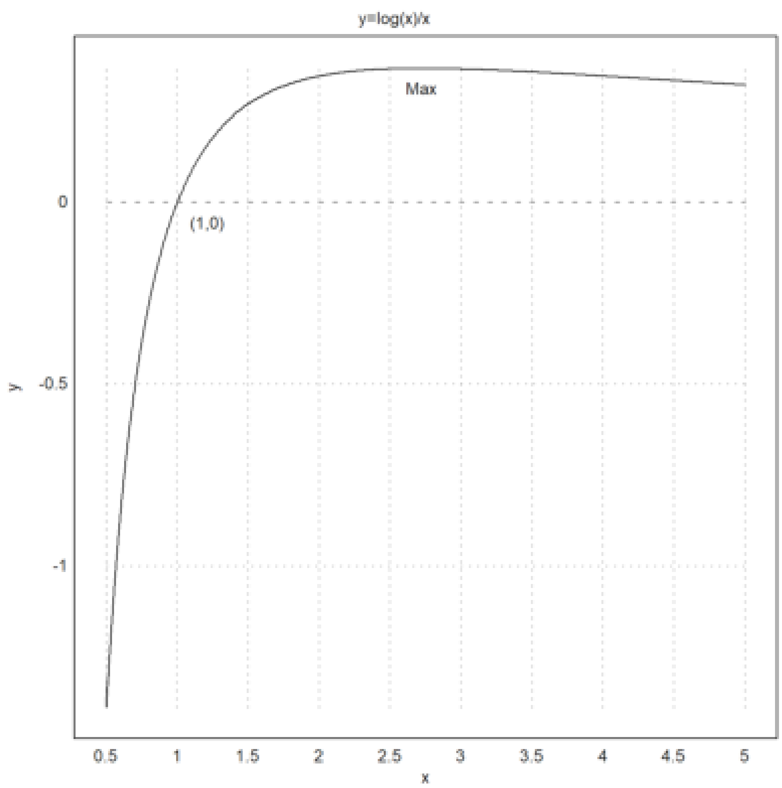
- judul dengan title = "..."
- Label x dan y dengan xl="...", yl="..."
- label teks lain dengan label("...",x,y)

Perintah label akan memplot ke dalam plot saat ini pada koordinat plot (x,y). Perintah ini dapat menerima argumen posisi.

```
>plot2d("x^3-x",-1,2,title="y=x^3-x",yl="y",xl="x"):
```

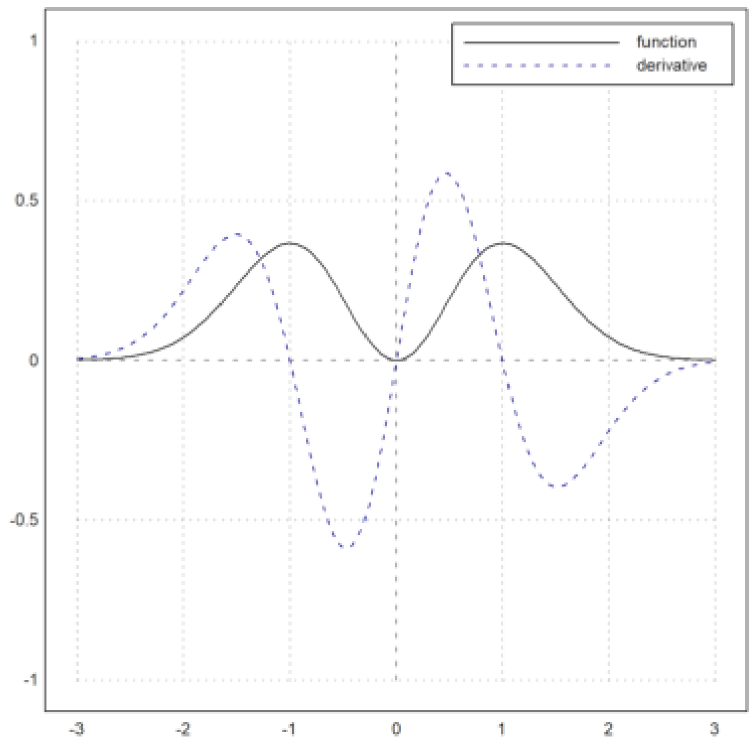



```
>expr := "log(x)/x"; ...
>  plot2d(expr,0.5,5,title="y="+expr,xl="x",yl="y"); ...
>  label("(1,0)",1,0); label("Max",E,expr(E),pos="lc"):
```



Ada juga fungsi labelbox(), yang dapat menampilkan fungsi dan teks. Fungsi ini membutuhkan vektor string dan warna, satu item untuk setiap fungsi.

```
>function f(x) &= x^2*exp(-x^2);  ...
>plot2d(&f(x),a=-3,b=3,c=-1,d=1);  ...
>plot2d(&diff(f(x),x),>add,color=blue,style="--"); ...
>labelbox(["function","derivative"],styles=["-","--"], ...
>   colors=[black,blue],w=0.4):
```

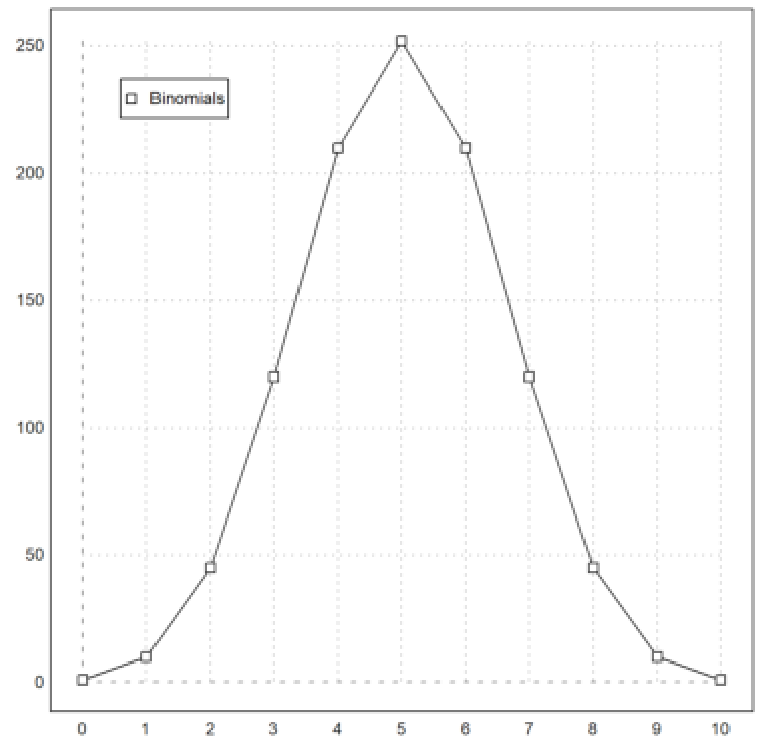


Kotak tersebut berlabuh di kanan atas secara default, tetapi >kiri menambatkannya di kiri atas. Anda dapat memindahkannya ke tempat mana pun yang Anda suka. Posisi jangkar adalah sudut kanan atas kotak, dan angkanya adalah pecahan dari ukuran jendela grafik. Lebarnya adalah otomatis.

Untuk plot titik, kotak label juga dapat digunakan. Tambahkan sebuah parameter >titik, atau sebuah vektor bendera, satu untuk setiap label.

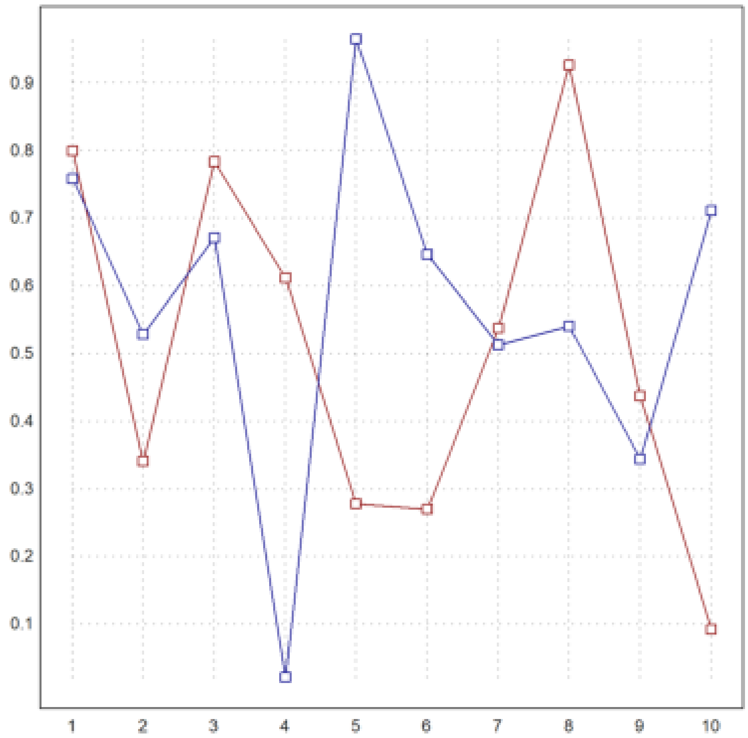
Pada contoh berikut ini, hanya ada satu fungsi. Jadi kita dapat menggunakan string dan bukan vektor string. Kami mengatur warna teks menjadi hitam untuk contoh ini.

```
>n=10; plot2d(0:n,bin(n,0:n),>addpoints); ...
>labelbox("Binomials",styles="[]",>points,x=0.1,y=0.1, ...
>tcolor=black,>left):
```



Gaya plot ini juga tersedia di statplot(). Seperti pada plot2d() warna dapat diatur untuk setiap baris plot. Terdapat lebih banyak plot khusus untuk keperluan statistik (lihat tutorial tentang statistik).

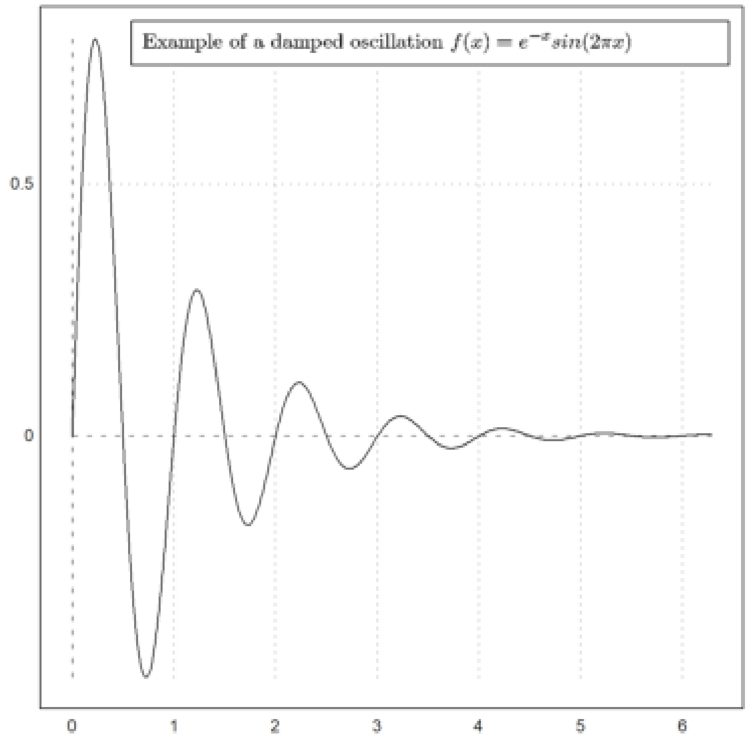
```
>statplot(1:10,random(2,10),color=[red,blue]):
```



Fitur yang serupa adalah fungsi textbox().

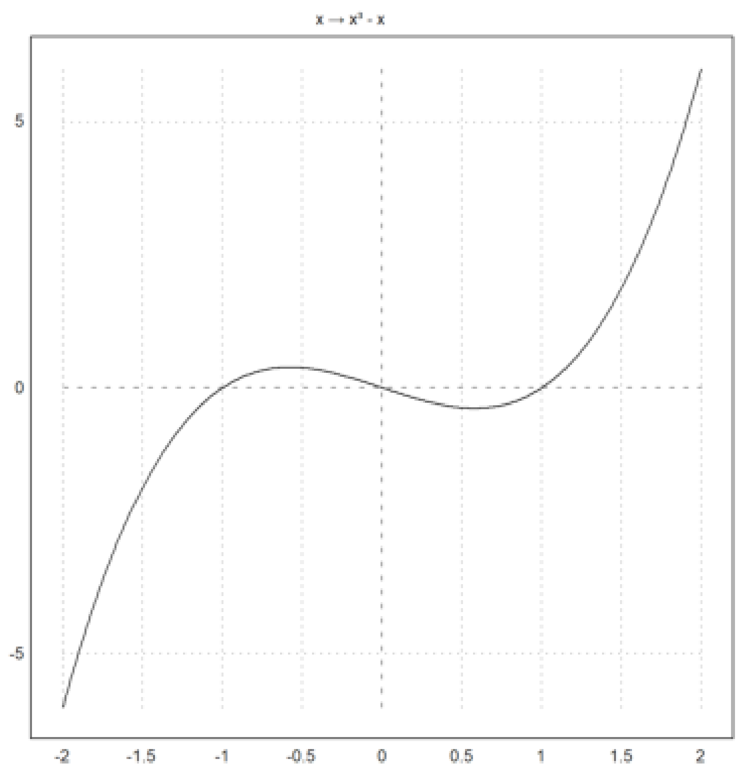
Lebarnya secara default adalah lebar maksimal baris teks. Tetapi, ini juga dapat diatur oleh pengguna.

```
>function f(x) &= exp(-x)*sin(2*pi*x); ...
>plot2d("f(x)",0,2pi); ...
>textbox(latex("\text{Example of a damped oscillation}\ f(x)=e^{-x}sin(2\pi
```



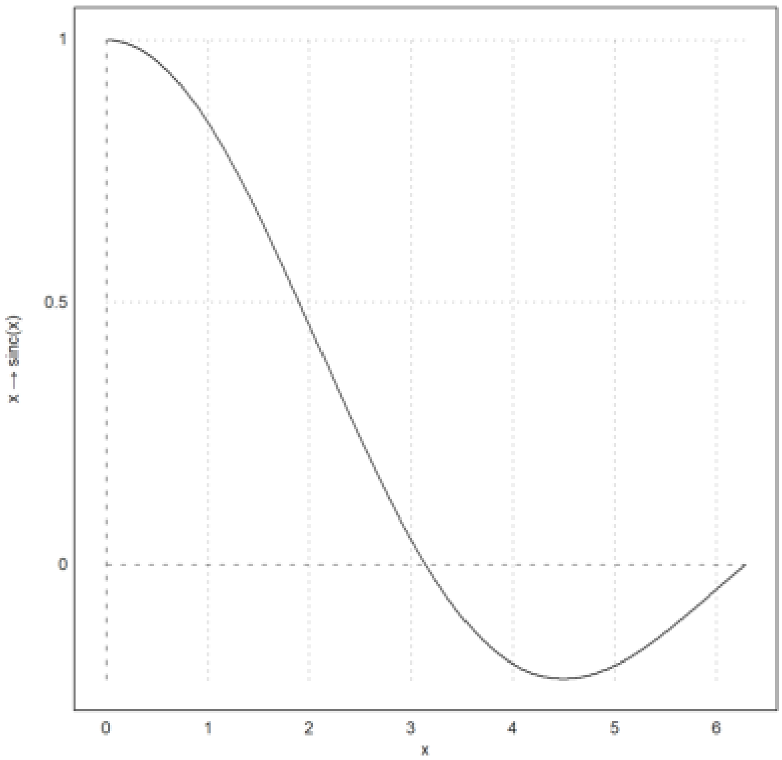
Label teks, judul, kotak label, dan teks lainnya dapat berisi string Unicode (lihat sintaks EMT untuk mengetahui lebih lanjut tentang string Unicode).

```
>plot2d("x^3-x",title=u"x &rarr; x&sup3; - x"):
```



Label pada sumbu x dan y bisa vertikal, begitu juga dengan sumbu.

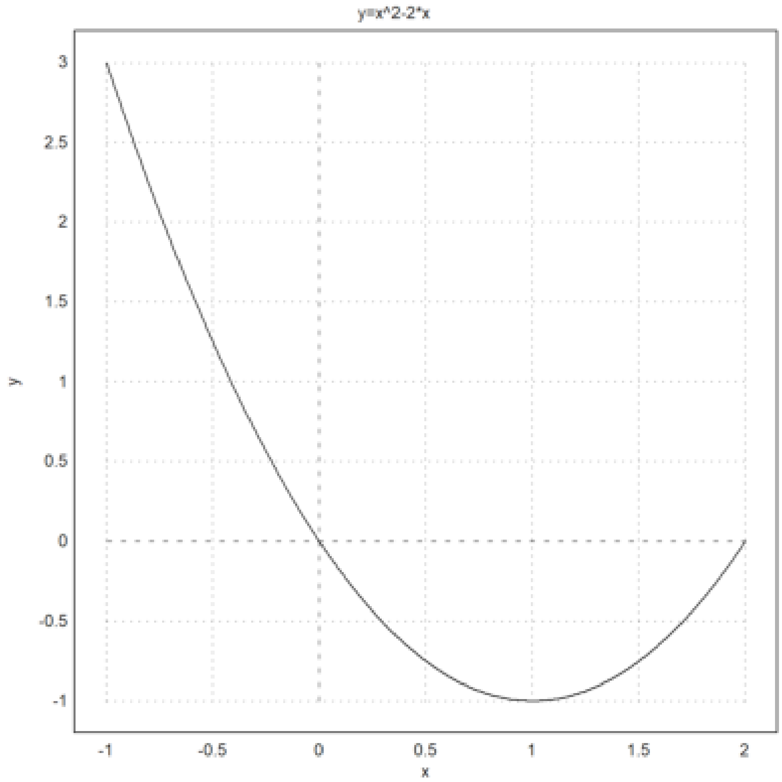
```
>plot2d("sinc(x)",0,2pi,xl="x",yl=u"x &rarr; sinc(x)",>vertical):
```



Contoh Tambahan
1. Buat plot dari fungsi tersebut.

$$y = x^2 - 2x$$

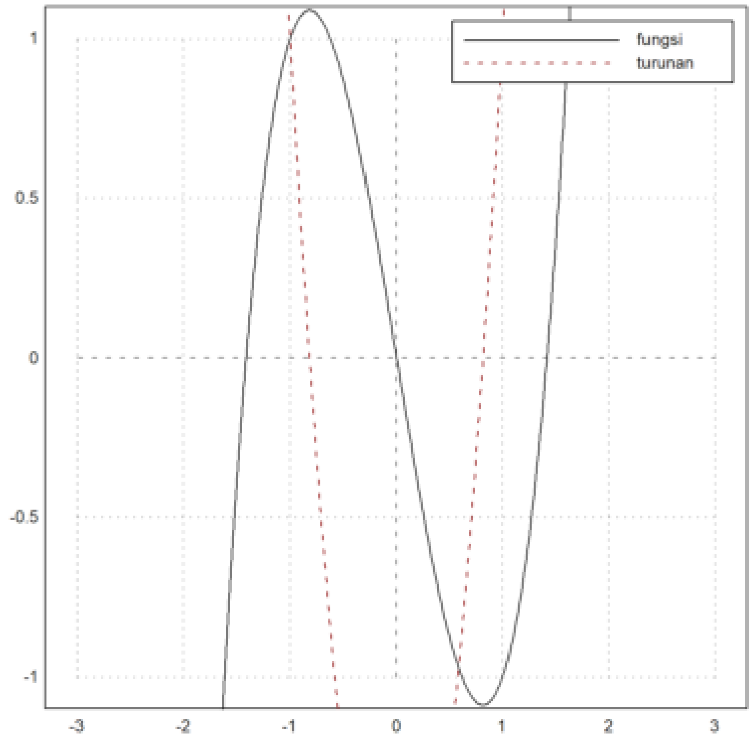
```
>plot2d("x^2-2*x",-1,2,title="y=x^2-2*x",yl="y",xl="x"):
```



2. Buatlah plot dari fungsi tersebut dan turunannya!

$$f(x) = x^3 - 2x$$

```
>function f(x)&=x^3-2*x; ...
>plot2d(&f(x),a=-3,b=3,c=-1,d=1); ...
>plot2d(&diff(f(x),x),>add,color=red,style="--"); ...
>labelbox(["fungsi", "turunan"],styles=["-","--"], ...
>colors=[black,red],w=0.4):
```




```
>
```

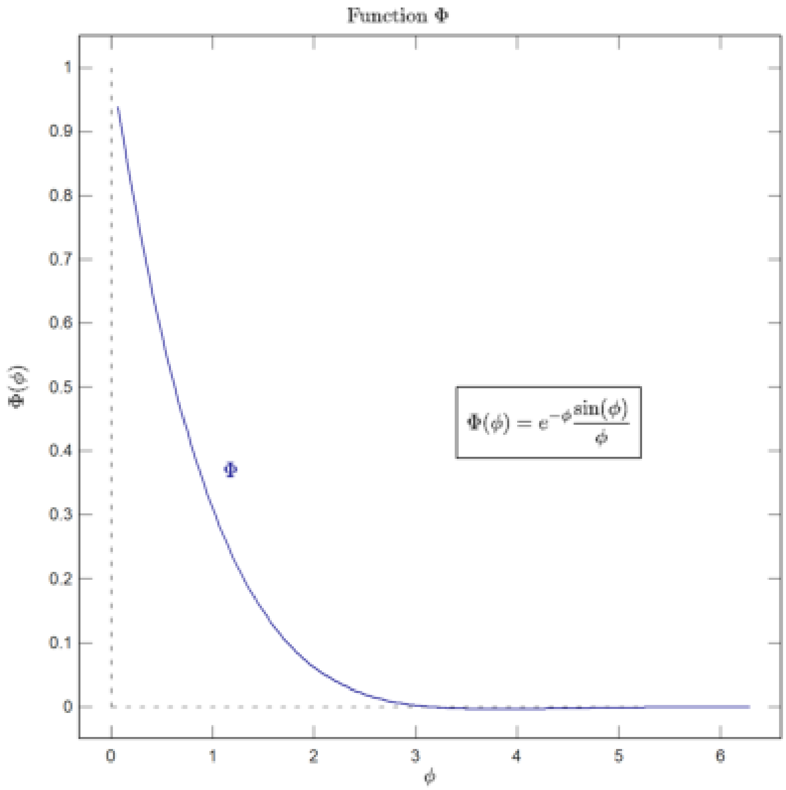
LaTeX

Anda juga dapat memplot formula LaTeX jika Anda telah menginstal sistem LaTeX. Saya merekomendasikan MiKTeX. Jalur ke binari "lateks" dan "dvipng" harus berada di jalur sistem, atau Anda harus mengatur LaTeX di menu opsi.

Perlu diperhatikan bahwa penguraian LaTeX berjalan lambat. Jika Anda ingin menggunakan LaTeX dalam plot animasi, Anda harus memanggil latex() sebelum perulangan sekali dan menggunakan hasilnya (gambar dalam matriks RGB).

Pada plot berikut ini, kita menggunakan LaTeX untuk label x dan y, sebuah label, sebuah kotak label, dan judul plot.

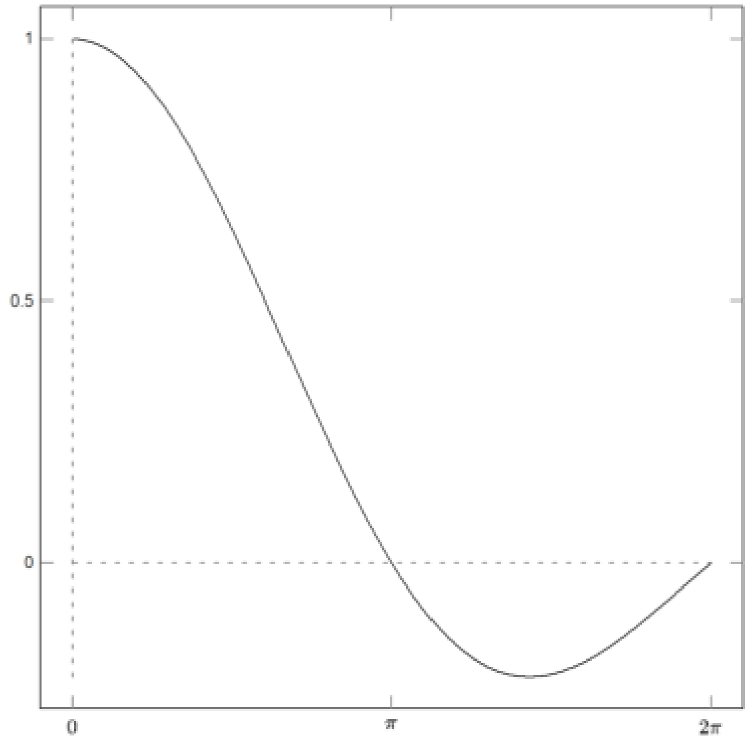
```
>plot2d("exp(-x)*sin(x)/x",a=0,b=2pi,c=0,d=1,grid=6,color=blue, ...
>  title=latex("\text{Function $\Phi$}"), ...
>  xl=latex("\phi"),yl=latex("\Phi(\phi)")); ...
>textbox( ...
>  latex("\Phi(\phi) = e^{-\phi} \frac{\sin(\phi)}{\phi}"),x=0.8,y=0.5); ...
>label(latex("\Phi",color=blue),1,0.4):
```



Seringkali, kita menginginkan spasi dan label teks yang tidak sesuai pada sumbu x. Kita dapat menggunakan xaxis() dan yaxis() seperti yang akan kita tunjukkan nanti.

Cara termudah adalah dengan membuat plot kosong dengan sebuah frame menggunakan grid=4, dan kemudian menambahkan grid dengan ygrid() dan xgrid(). Pada contoh berikut, kita menggunakan tiga buah string LaTeX untuk label pada sumbu x dengan xtick().

```
>plot2d("sinc(x)",0,2pi,grid=4,<ticks); ...
>ygrid(-2:0.5:2,grid=6); ...
>xgrid([0:2]*pi,<ticks,grid=6);  ...
>xtick([0,pi,2pi],["0","\pi","2\pi"],>latex):
```



Tentu saja, fungsi juga dapat digunakan.

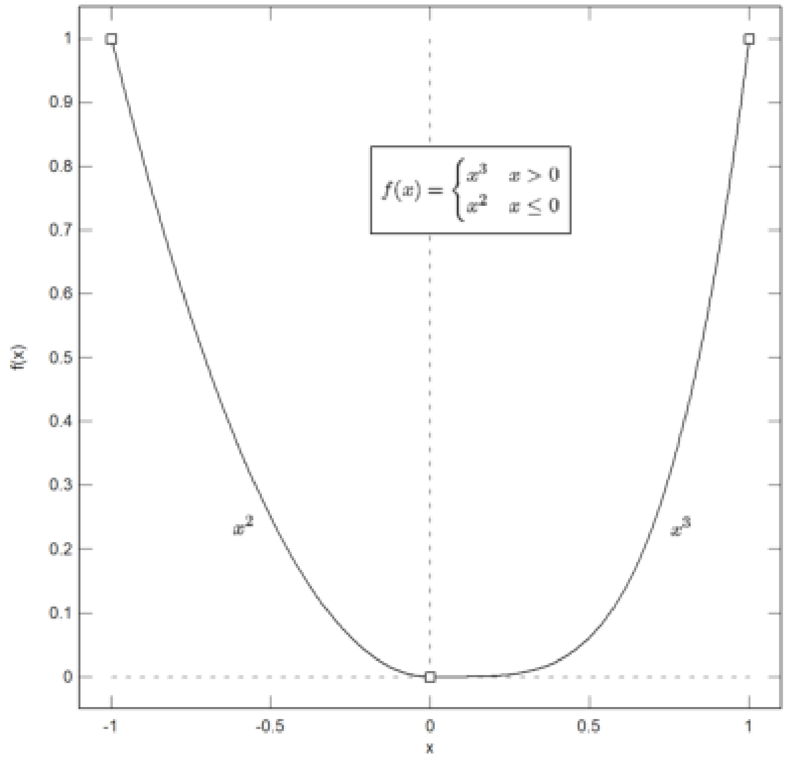
```
>function map f(x)
```

```
  if x>0 then return x^4
  else return x^2
  endif
  endfunction
```

Parameter "map" membantu menggunakan fungsi untuk vektor. Untuk plot, hal ini tidak diperlukan. Tetapi untuk menunjukkan bahwa vektorisasi berguna, kami menambahkan beberapa titik kunci pada plot pada x =-1, x = 0 dan x = 1.

Pada plot berikut, kita juga memasukkan beberapa kode LaTeX. Kami menggunakannya untuk dua label dan sebuah kotak teks. Tentu saja, Anda hanya dapat menggunakan LaTeX jika Anda telah menginstal LaTeX dengan benar.

```
>plot2d("f",-1,1,xl="x",yl="f(x)",grid=6);  ...
>plot2d([-1,0,1],f([-1,0,1]),>points,>add); ...
>label(latex("x^3"),0.72,f(0.72)); ...
>label(latex("x^2"),-0.52,f(-0.52),pos="ll"); ...
>textbox( ...
>  latex("f(x)=\begin{cases} x^3 & x>0 \\ x^2 & x \le 0\end{cases}"), ...
>  x=0.7,y=0.2):
```



Interaksi Pengguna

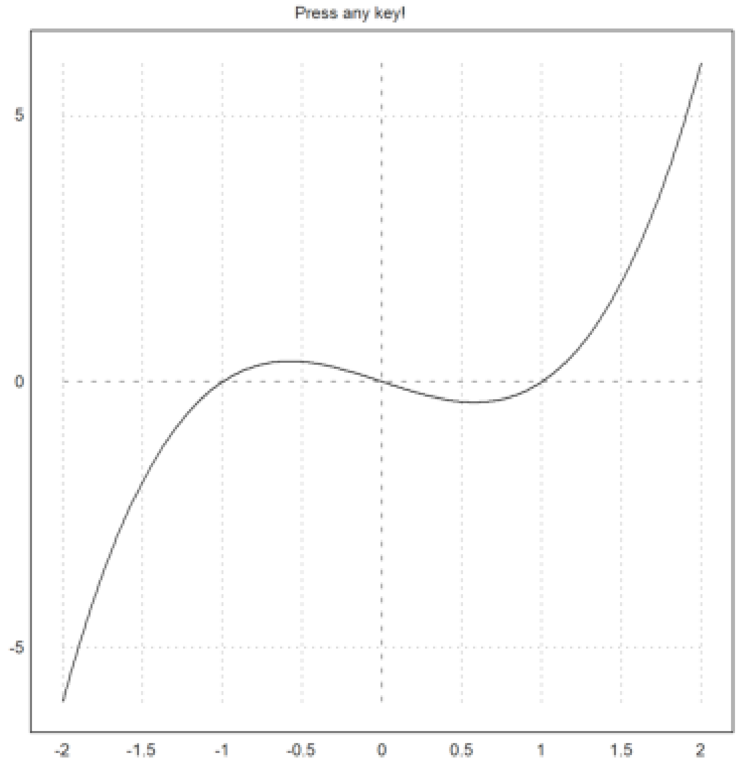
Ketika memplot fungsi atau ekspresi, parameter >user memungkinkan pengguna untuk memperbesar dan menggeser plot dengan tombol kursor atau mouse. Pengguna dapat

- zoom dengan + atau -
- memindahkan plot dengan tombol kursor
- pilih jendela plot dengan mouse
- mengatur ulang tampilan dengan spasi
- keluar dengan kembali

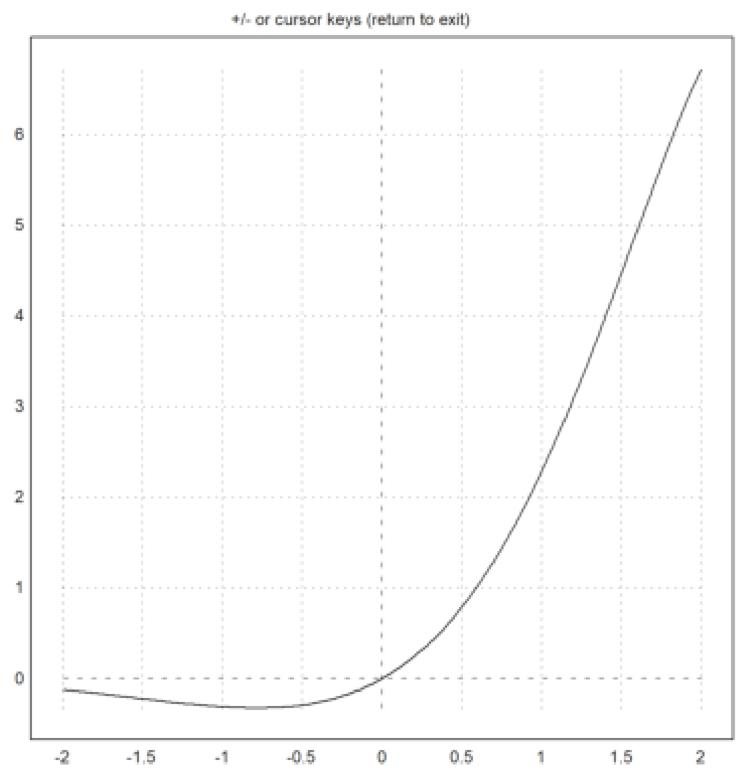
Tombol spasi akan mengatur ulang plot ke jendela plot asli.

Saat memplot data, bendera >user hanya akan menunggu penekanan tombol.

```
>plot2d({{"x^3-a*x",a=1}},>user,title="Press any key!"):
```




```
>plot2d("exp(x)*sin(x)",user=true, ...
>  title="+/- or cursor keys (return to exit)"):
```



Berikut ini menunjukkan cara interaksi pengguna tingkat lanjut (lihat tutorial mengenai pemrograman untuk detailnya).

Fungsi bawaan mousedrag() menunggu peristiwa mouse atau keyboard. Fungsi ini melaporkan mouse ke bawah, mouse bergerak atau mouse ke atas, dan penekanan tombol. Fungsi dragpoints() memanfaatkan hal ini, dan mengizinkan pengguna untuk menyeret titik manapun di dalam plot.

Kita membutuhkan fungsi plot terlebih dahulu. Sebagai contoh, kita melakukan interpolasi dalam 5 titik dengan polinomial. Fungsi ini harus memplot ke dalam area plot yang tetap.

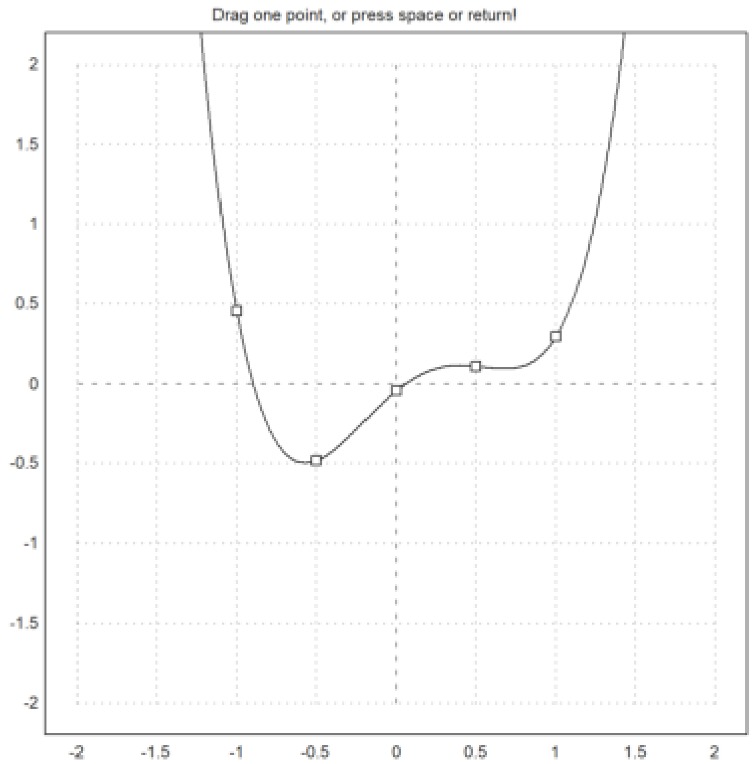
```
>function plotf(xp,yp,select) ...
```

```
    d=interp(xp,yp);
    plot2d("interpval(xp,d,x)";d,xp,r=2);
    plot2d(xp,yp,>points,>add);
    if select>0 then
      plot2d(xp[select],yp[select],color=red,>points,>add);
    endif;
    title("Drag one point, or press space or return!");
  endfunction
```

Perhatikan parameter titik koma pada plot2d (d dan xp), yang diteruskan ke evaluasi fungsi interp(). Tanpa ini, kita harus menulis fungsi plotinterp() terlebih dahulu, untuk mengakses nilai secara global.

Sekarang kita menghasilkan beberapa nilai acak, dan membiarkan pengguna menyeret titik-titiknya.

```
>t=-1:0.5:1; dragpoints("plotf",t,random(size(t))-0.5):
```



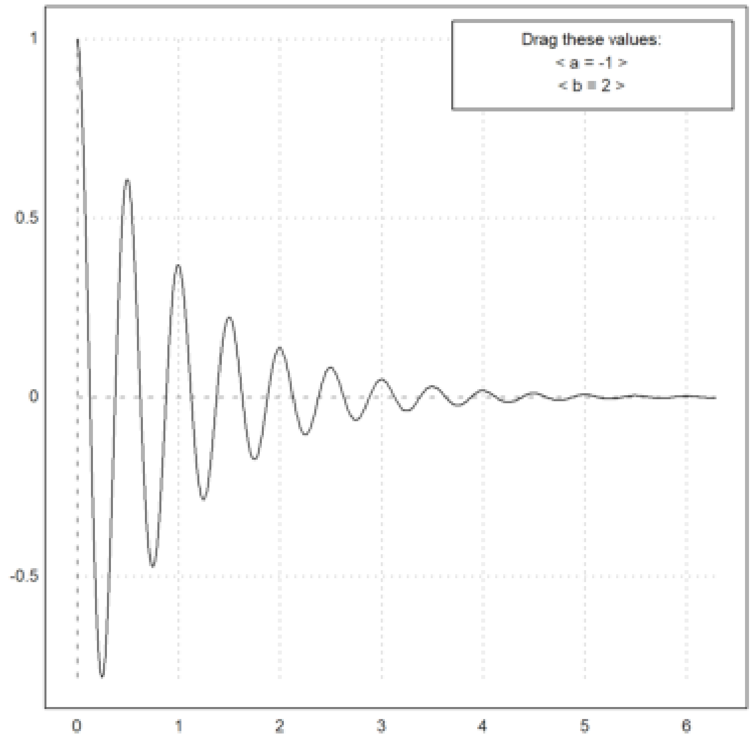
Ada juga fungsi yang memplot fungsi lain tergantung pada vektor parameter, dan memungkinkan pengguna menyesuaikan parameter ini.

Pertama, kita memerlukan fungsi plot.

```
>function plotf([a,b]) := plot2d("exp(a*x)*cos(2pi*b*x)",0,2pi;a,b);
```

Kemudian kita membutuhkan nama untuk parameter, nilai awal dan matriks rentang nx2, dan secara opsional, sebuah garis judul. Terdapat slider interaktif, yang dapat mengatur nilai oleh pengguna. Fungsi dragvalues() menyediakan ini.

```
>dragvalues("plotf",["a","b"],[-1,2],[[-2,2];[1,10]], ...
>  heading="Drag these values:",hcolor=black):
```



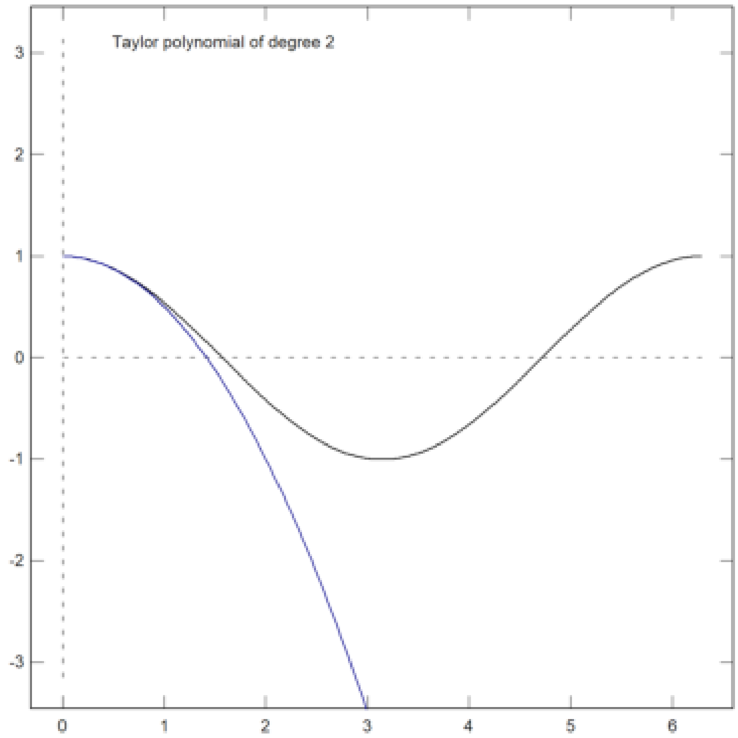
Anda dapat membatasi nilai yang diseret menjadi bilangan bulat. Sebagai contoh, kita menulis fungsi plot, yang memplot polinomial Taylor dengan derajat n ke fungsi kosinus.

```
>function plotf(n) ...
```

```
  plot2d("cos(x)",0,2pi,>square,grid=6);
  plot2d(&"taylor(cos(x),x,0,@n)",color=blue,>add);
  textbox("Taylor polynomial of degree "+n,0.1,0.02,style="t",>left);
  endfunction
```

Sekarang kita membiarkan derajat n bervariasi dari 0 sampai 20 dalam 20 stop. Hasil dari dragvalues() digunakan untuk memplot sketsa dengan n ini, dan untuk menyisipkan plot ke dalam buku catatan.

```
>nd=dragvalues("plotf","degree",2,[0,20],20,y=0.8, ...
>   heading="Drag the value:"); ...
>plotf(nd):
```

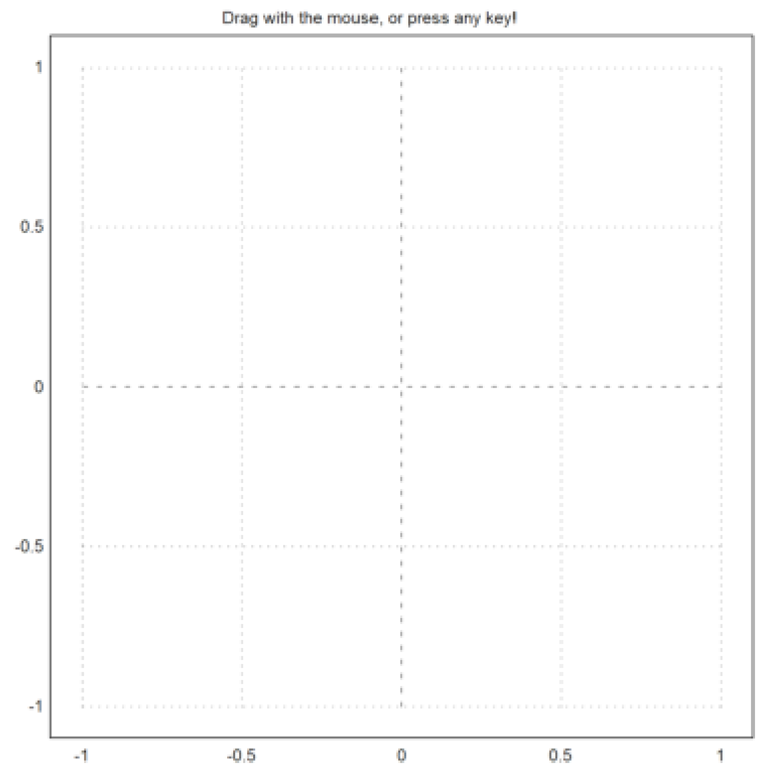


Berikut ini adalah peragaan sederhana dari fungsi ini. Pengguna dapat menggambar di atas jendela plot, meninggalkan jejak titik.

```
>function dragtest ...
```

```
    plot2d(none,r=1,title="Drag with the mouse, or press any key!");
    start=0;
    repeat
      {flag,m,time}=mousedrag();
      if flag==0 then return; endif;
      if flag==2 then
        hold on; mark(m[1],m[2]); hold off;
      endif;
    end
  endfunction
```

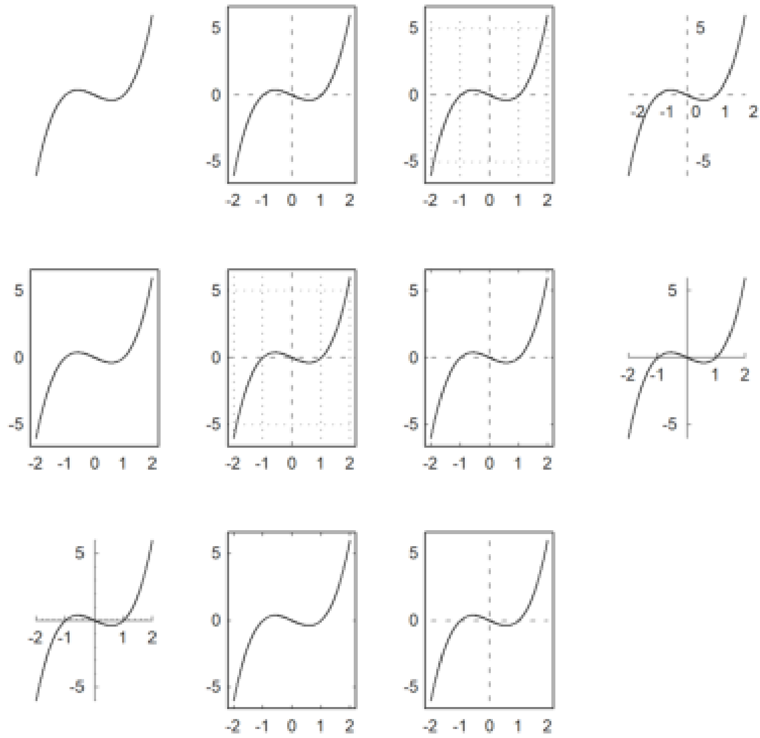
```
>dragtest: // lihat hasilnya dan cobalah lakukan!
```



Gaya Plot 2D

Secara default, EMT menghitung tick sumbu otomatis dan menambahkan label pada setiap tick. Hal ini dapat diubah dengan parameter kisi-kisi. Gaya default sumbu dan label dapat dimodifikasi. Selain itu, label dan judul dapat ditambahkan secara manual. Untuk mengatur ulang ke gaya default, gunakan reset().

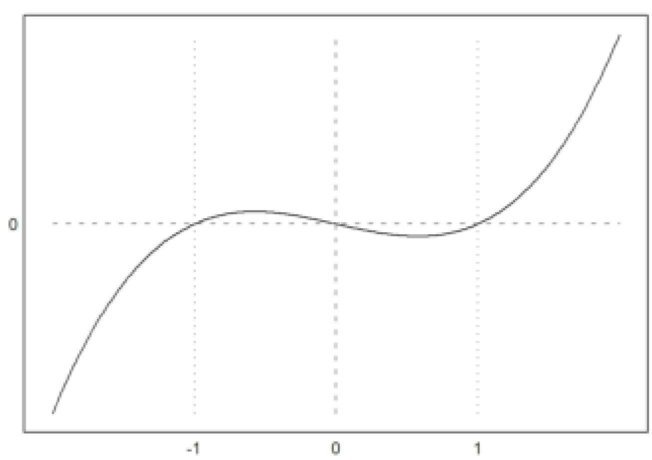
```
>aspect();
>figure(3,4); ...
> figure(1); plot2d("x^3-x",grid=0); ... // no grid, frame or axis
> figure(2); plot2d("x^3-x",grid=1); ... // x-y-axis
> figure(3); plot2d("x^3-x",grid=2); ... // default ticks
> figure(4); plot2d("x^3-x",grid=3); ... // x-y- axis with labels inside
> figure(5); plot2d("x^3-x",grid=4); ... // no ticks, only labels
> figure(6); plot2d("x^3-x",grid=5); ... // default, but no margin
> figure(7); plot2d("x^3-x",grid=6); ... // axes only
> figure(8); plot2d("x^3-x",grid=7); ... // axes only, ticks at axis
> figure(9); plot2d("x^3-x",grid=8); ... // axes only, finer ticks at axis
> figure(10); plot2d("x^3-x",grid=9); ... // default, small ticks inside
> figure(11); plot2d("x^3-x",grid=10); ...// no ticks, axes only
> figure(0):
```



Parameter <frame mematikan bingkai, dan framecolor=blue menetapkan bingkai ke warna biru.

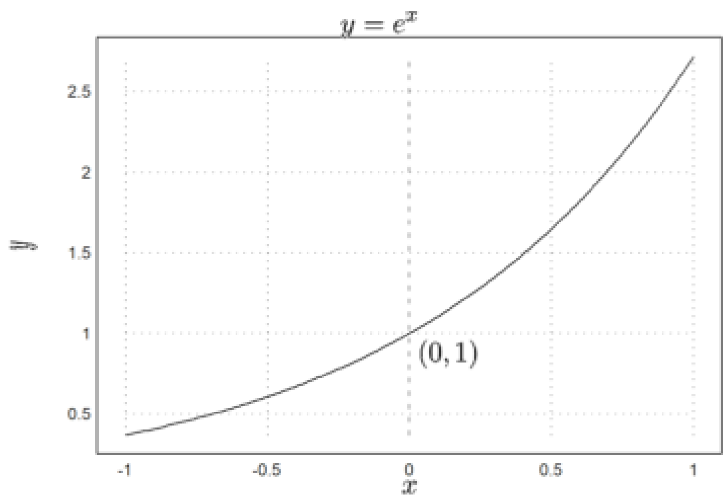
Jika Anda menginginkan tanda centang Anda sendiri, Anda dapat menggunakan style=0, dan menambahkan semuanya nanti.

```
>aspect(1.5);
>plot2d("x^3-x",grid=0); // plot
>frame; xgrid([-1,0,1]); ygrid(0): // add frame and grid
```



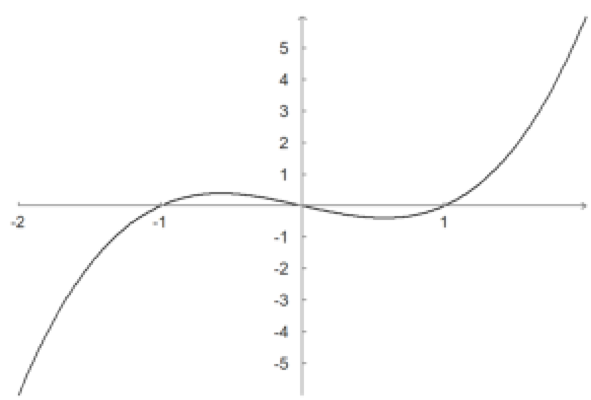
Untuk judul plot dan label sumbu, lihat contoh berikut.

```
>plot2d("exp(x)",-1,1);
>textcolor(black); // set the text color to black
>title(latex("y=e^x")); // title above the plot
>xlabel(latex("x")); // "x" for x-axis
>ylabel(latex("y"),>vertical); // vertical "y" for y-axis
>label(latex("(0,1)"),0,1,color=blue): // label a point
```



Sumbu dapat digambar secara terpisah dengan sumbu x() dan sumbu y().

```
>plot2d("x^3-x",<grid,<frame);
>xaxis(0,xx=-2:1,style="->"); yaxis(0,yy=-5:5,style="->"):
```

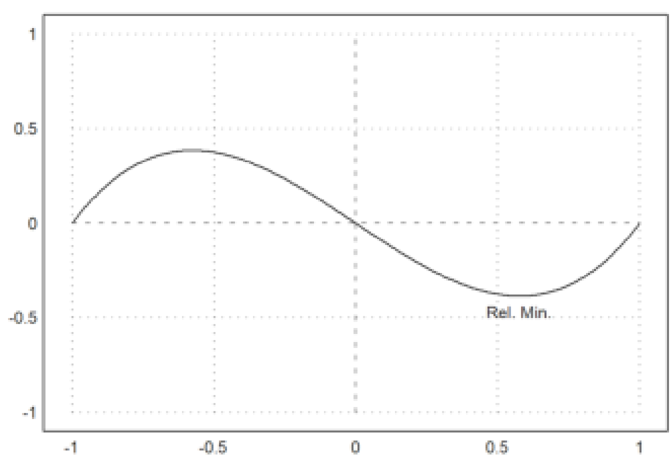


Teks pada plot dapat diatur dengan label(). Pada contoh berikut ini, "lc" berarti lower center. Ini mengatur posisi label relatif terhadap koordinat plot.

```
>function f(x) &= x^3-x
```

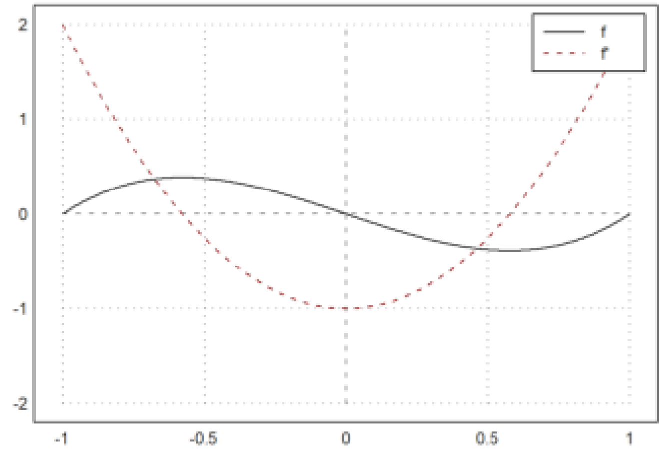
```
                                  3
                                 x  - x
```

```
>plot2d(f,-1,1,>square);
>x0=fmin(f,0,1); // compute point of minimum
>label("Rel. Min.",x0,f(x0),pos="lc"): // add a label there
```

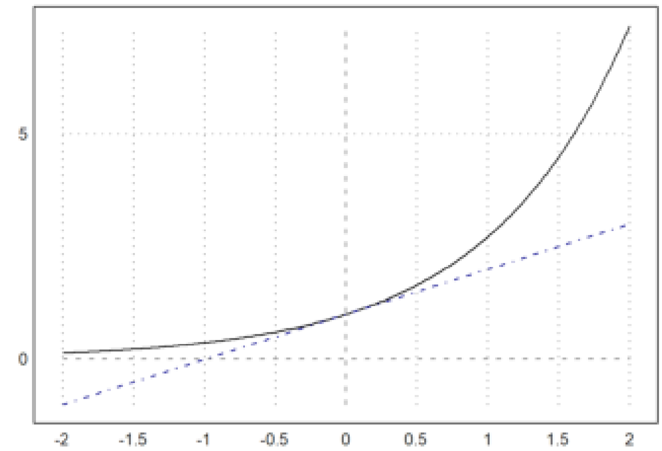


Terdapat juga kotak teks.

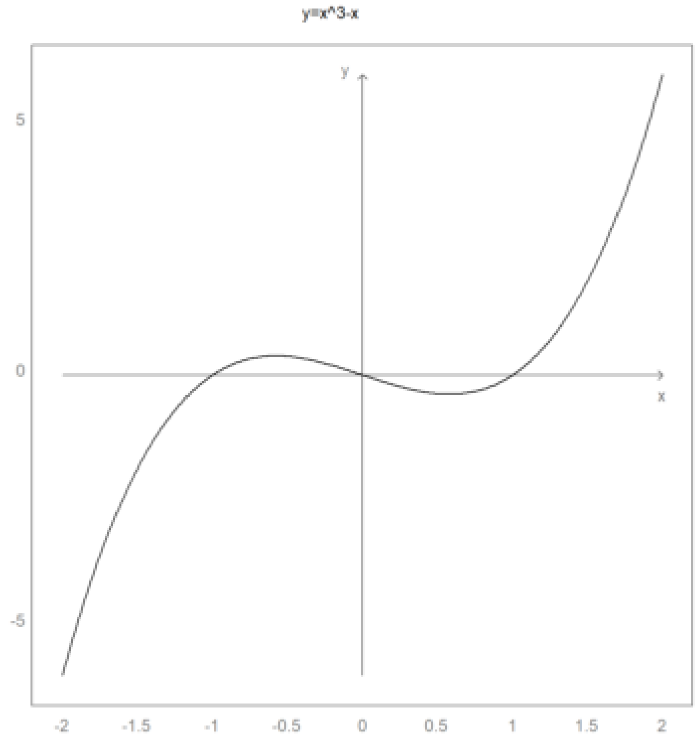
```
>plot2d(&f(x),-1,1,-2,2); // function
>plot2d(&diff(f(x),x),>add,style="--",color=red); // derivative
>labelbox(["f","f'"],["-","--"],[black,red]): // label box
```




```
>plot2d(["exp(x)","1+x"],color=[black,blue],style=["-","-.-"]):
```



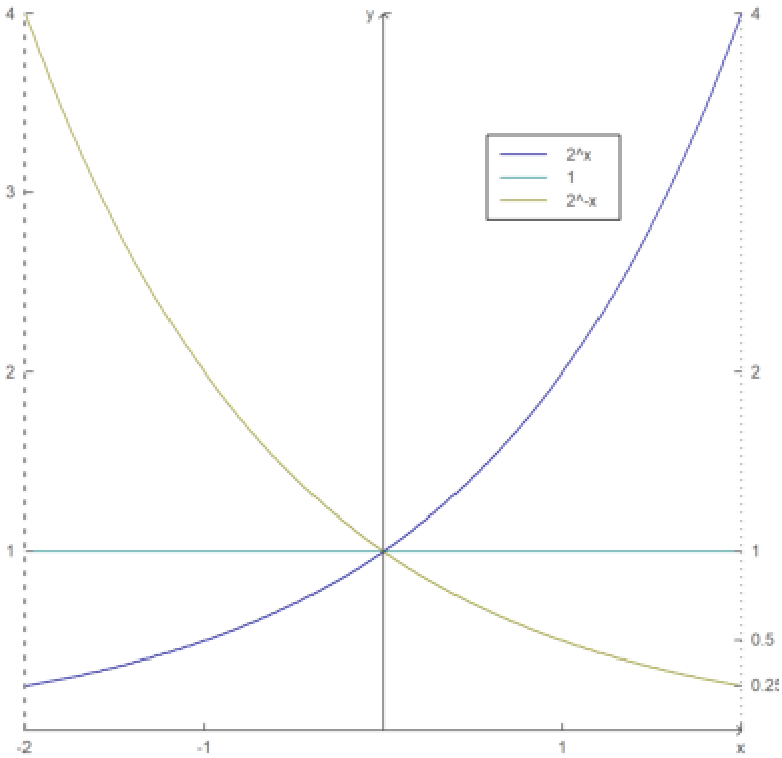

```
>gridstyle("->",color=gray,textcolor=gray,framecolor=gray);  ...
> plot2d("x^3-x",grid=1);   ...
> settitle("y=x^3-x",color=black); ...
> label("x",2,0,pos="bc",color=gray);  ...
> label("y",0,6,pos="cl",color=gray); ...
> reset():
```



Untuk kontrol yang lebih banyak lagi, sumbu x dan sumbu y dapat dilakukan secara manual.

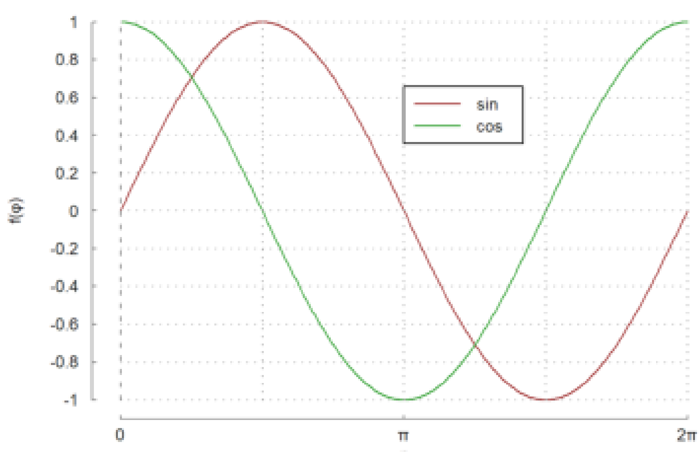
Perintah fullwindow() memperluas jendela plot karena kita tidak lagi membutuhkan tempat untuk label di luar jendela plot. Gunakan shrinkwindow() atau reset() untuk mengatur ulang ke default.

```
>fullwindow; ...
> gridstyle(color=darkgray,textcolor=darkgray); ...
> plot2d(["2^x","1","2^(-x)"],a=-2,b=2,c=0,d=4,<grid,color=4:6,<frame); ...
> xaxis(0,-2:1,style="->"); xaxis(0,2,"x",<axis); ...
> yaxis(0,4,"y",style="->"); ...
> yaxis(-2,1:4,>left); ...
> yaxis(2,2^(-2:2),style=".",<left); ...
> labelbox(["2^x","1","2^-x"],colors=4:6,x=0.8,y=0.2); ...
> reset:
```



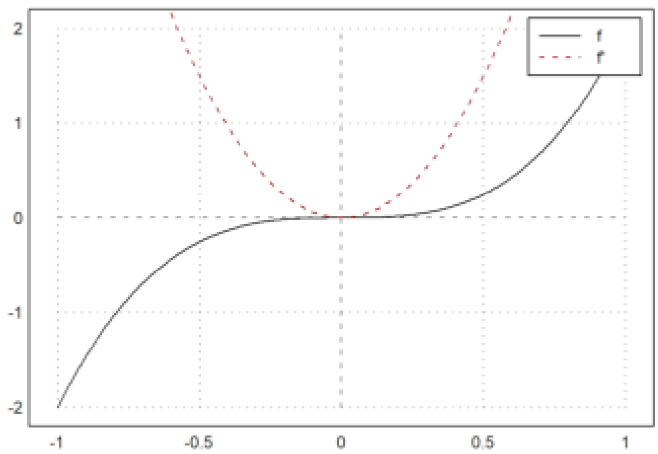
Berikut ini adalah contoh lain, di mana string Unicode digunakan dan sumbu di luar area plot.

```
>aspect(1.5);
>plot2d(["sin(x)","cos(x)"],0,2pi,color=[red,green],<grid,<frame); ...
> xaxis(-1.1,(0:2)*pi,xt=["0",u"&pi;",u"2&pi;"],style="-",>ticks,>zero);   .
> xgrid((0:0.5:2)*pi,<ticks); ...
> yaxis(-0.1*pi,-1:0.2:1,style="-",>zero,>grid); ...
> labelbox(["sin","cos"],colors=[red,green],x=0.5,y=0.2,>left); ...
> xlabel(u"&phi;"); ylabel(u"f(&phi;)"):
```



Contoh Tambahan:

```
>plot2d(&(2*x^3),-1,1,-2,2);
>plot2d(&diff(2*x^3,x),>add,style="--",color=red);
>labelbox(["f","f'"],["-","--"],[black,red]):
```



Memplot Data 2D

Jika x dan y adalah vektor data, data ini akan digunakan sebagai koordinat x dan y dari sebuah kurva. Dalam hal ini, a, b, c, dan d, atau radius r dapat ditentukan, atau jendela plot akan menyesuaikan secara otomatis dengan data. Atau, >square dapat diatur untuk mempertahankan rasio aspek persegi.

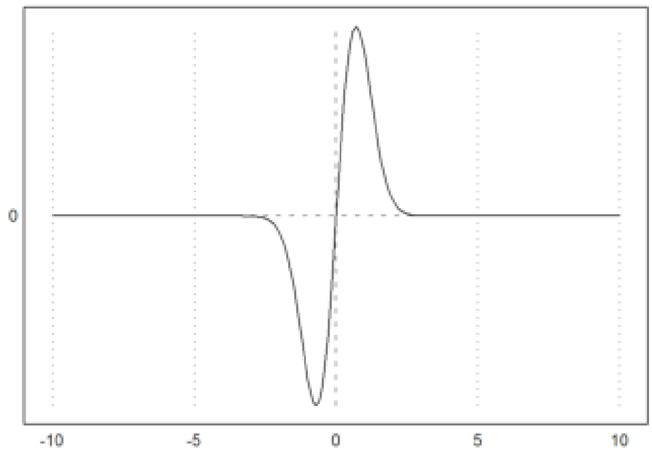
Memplot ekspresi hanyalah singkatan dari plot data. Untuk plot data, Anda memerlukan satu atau lebih baris nilai x, dan satu atau lebih baris nilai y. Dari rentang dan nilai x, fungsi plot2d akan menghitung data untuk diplot, secara default dengan evaluasi adaptif dari fungsi tersebut. Untuk plot titik, gunakan ">points", untuk garis dan titik campuran, gunakan ">addpoints".

Tetapi Anda dapat memasukkan data secara langsung.

- Gunakan vektor baris untuk x dan y untuk satu fungsi.
- Matriks untuk x dan y diplot baris demi baris.

Berikut adalah contoh dengan satu baris untuk x dan y.

```
>x=-10:0.1:10; y=exp(-x^2)*x; plot2d(x,y):
```

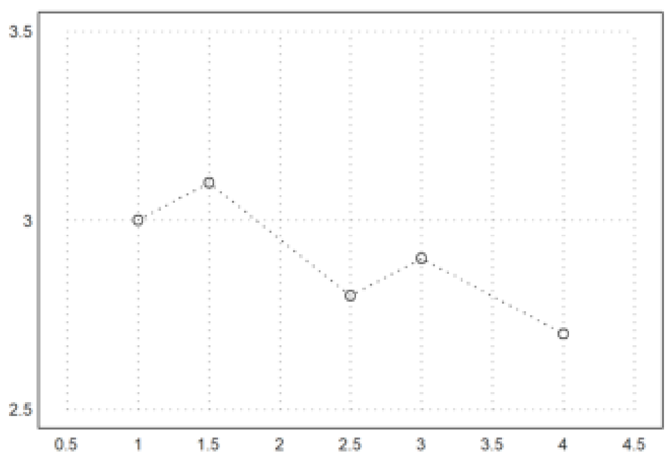


Data juga dapat diplot sebagai titik. Gunakan poin=true untuk ini. Plot ini berfungsi seperti poligon, namun hanya menggambar sudut-sudutnya saja.

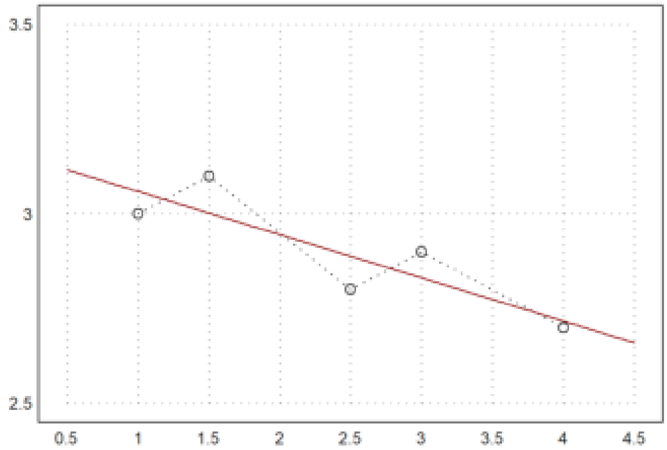
- style = "...": Pilih dari "[]", "<>", "o", ".", "..", "+", "*", "[]", "<>", "o", "..", "", "|".

Untuk memplot kumpulan titik, gunakan >titik. Jika warna adalah vektor warna, setiap titik mendapatkan warna yang berbeda. Untuk matriks koordinat dan vektor kolom, warna berlaku untuk baris matriks.
Parameter >addpoints menambahkan titik ke segmen garis untuk plot data.

```
>xdata=[1,1.5,2.5,3,4]; ydata=[3,3.1,2.8,2.9,2.7]; // data
>plot2d(xdata,ydata,a=0.5,b=4.5,c=2.5,d=3.5,style="."); // lines
>plot2d(xdata,ydata,>points,>add,style="o"): // add points
```

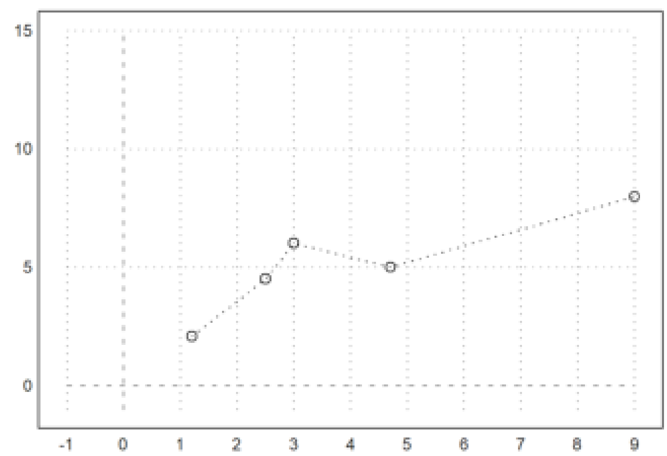



```
>p=polyfit(xdata,ydata,1); // get regression line
>plot2d("polyval(p,x)",>add,color=red): // add plot of line
```



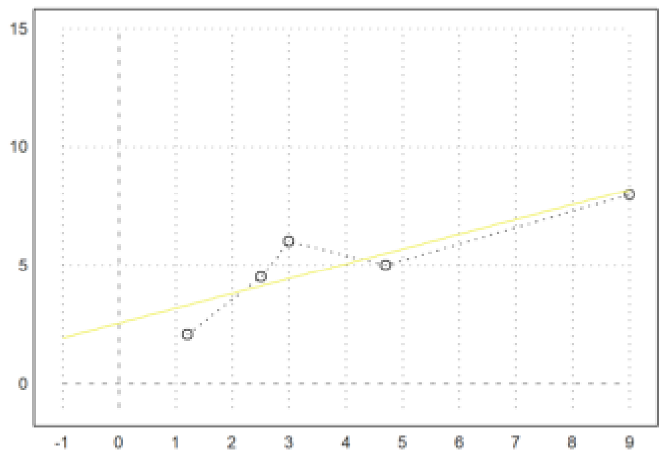
Contoh tambahan
1. Buatlah plot dari kumpulan titik berikut.

```
>xdata=[1.2,2.5,3,4.7,9]; ydata=[2.1,4.5,6,5,8]; //data
>plot2d(xdata,ydata,a=-1,b=9,c=-1,d=15,style=".");
>plot2d(xdata,ydata,>points,>add,style="o"):
```



Lalu buatlah garis regresinya!

```
>p=polyfit(xdata,ydata,1);
>plot2d("polyval(p,x)",>add,color=yellow):
```



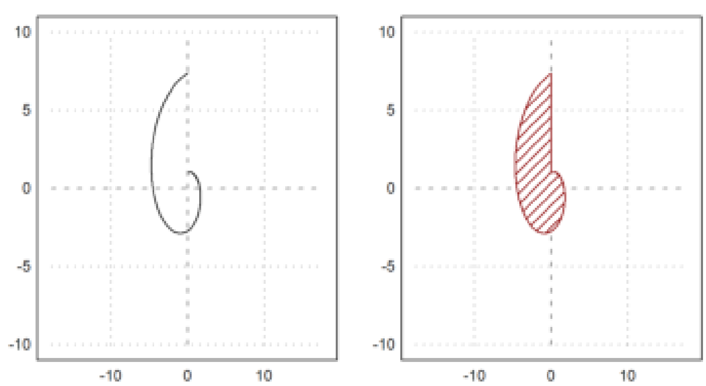
Menggambar Daerah Yang Dibatasi Kurva

Plot data benar-benar berupa poligon. Kita juga dapat memplot kurva atau kurva terisi.

-filled=true mengisi plot.
- style="...": Pilih dari "", "/", "\", "\/".
-fillcolor: Lihat di atas untuk warna yang tersedia.

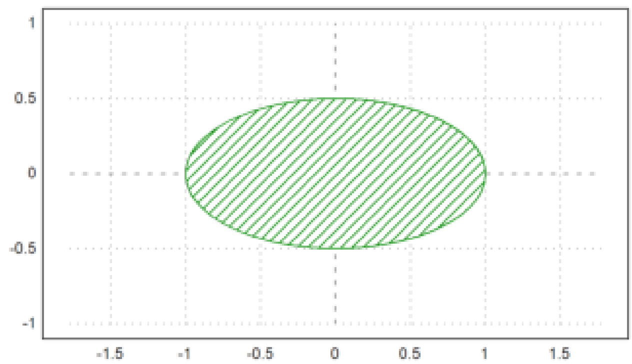
Warna isian ditentukan oleh argumen "fillcolor", dan pada opsional <outline mencegah menggambar garis batas untuk semua gaya kecuali yang default.

```
>t=linspace(0,2pi,1000); // parameter for curve
>x=sin(t)*exp(t/pi); y=cos(t)*exp(t/pi); // x(t) and y(t)
>figure(1,2); aspect(16/9)
>figure(1); plot2d(x,y,r=10); // plot curve
>figure(2); plot2d(x,y,r=10,>filled,style="/",fillcolor=red); // fill curve
>figure(0):
```

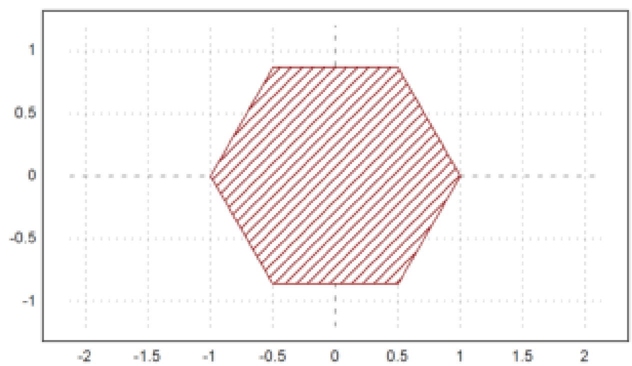


Pada contoh berikut ini, kami memplot elips yang terisi dan dua segi enam yang terisi menggunakan kurva tertutup dengan 6 titik dengan gaya isian yang berbeda.

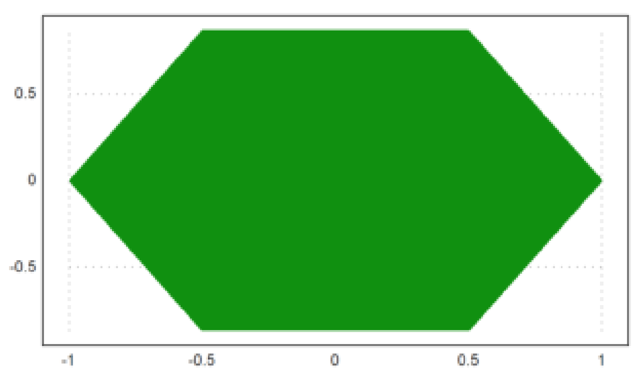
```
>x=linspace(0,2pi,1000); plot2d(sin(x),cos(x)*0.5,r=1,>filled,style="/"):
```



```
>t=linspace(0,2pi,6); ...
>plot2d(cos(t),sin(t),>filled,style="/",fillcolor=red,r=1.2):
```

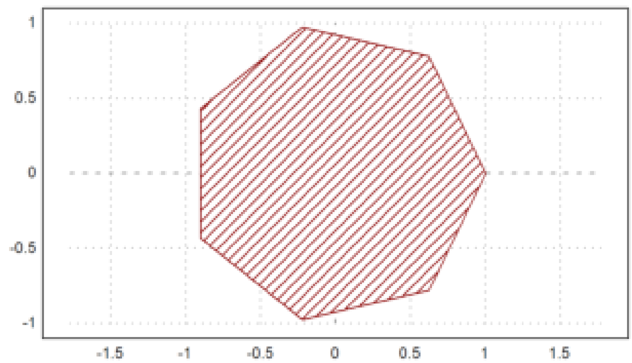


```
>t=linspace(0,2pi,6); plot2d(cos(t),sin(t),>filled,style="#"):
```



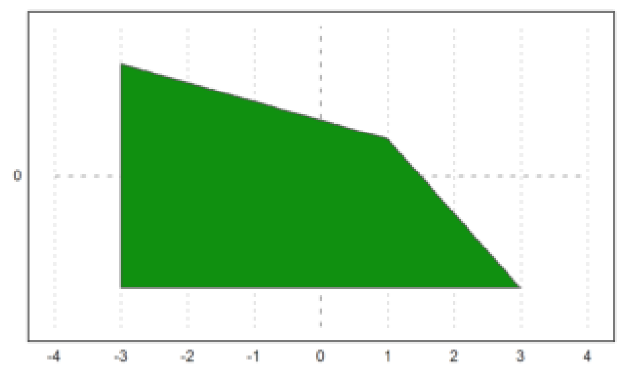
Contoh lainnya adalah septagon, yang kita buat dengan 7 titik pada lingkaran satuan.

```
>t=linspace(0,2pi,7);  ...
> plot2d(cos(t),sin(t),r=1,>filled,style="/",fillcolor=red):
```



Berikut ini adalah himpunan nilai maksimal dari empat kondisi linier yang kurang dari atau sama dengan 3. Ini adalah A[k].v<=3 untuk semua barisan A. Untuk mendapatkan sudut-sudut yang bagus, kita menggunakan n yang relatif besar.

```
>A=[2,1;1,2;-1,0;0,-1];
>function f(x,y) := max([x,y].A');
>plot2d("f",r=4,level=[0;3],color=green,n=111):
```

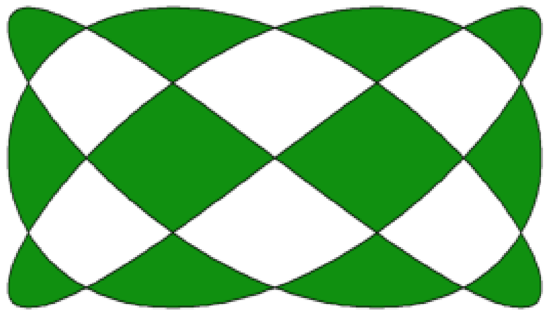


Poin utama dari bahasa matriks adalah bahwa bahasa ini memungkinkan untuk menghasilkan tabel fungsi dengan mudah

```
>t=linspace(0,2pi,1000); x=cos(3*t); y=sin(4*t);
```

Kita sekarang memiliki vektor nilai x dan y. plot2d() dapat memplot nilai-nilai ini sebagai sebuah kurva yang menghubungkan titik-titik. Plot dapat diisi. Dalam kasus ini, hal ini memberikan hasil yang bagus karena aturan penggulungan, yang digunakan untuk isi.

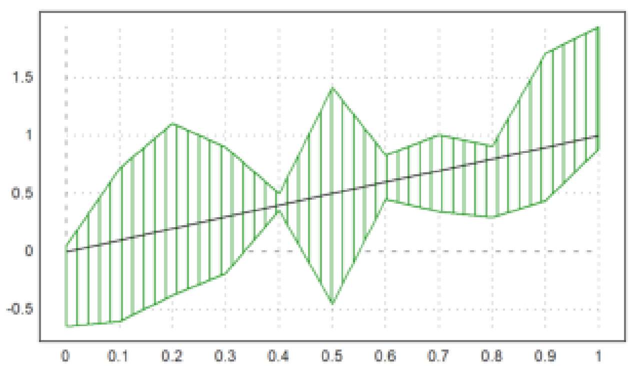
```
>plot2d(x,y,<grid,<frame,>filled):
```



Vektor interval diplot terhadap nilai x sebagai wilayah yang terisi antara nilai bawah dan atas interval.

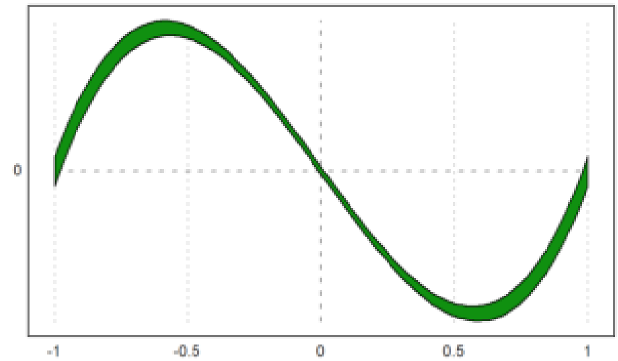
Hal ini dapat berguna untuk memplot kesalahan perhitungan. Tetapi juga dapat digunakan untuk memplot kesalahan statistik.

```
>t=0:0.1:1; ...
> plot2d(t,interval(t-random(size(t)),t+random(size(t))),style="|"); ...
> plot2d(t,t,add=true):
```



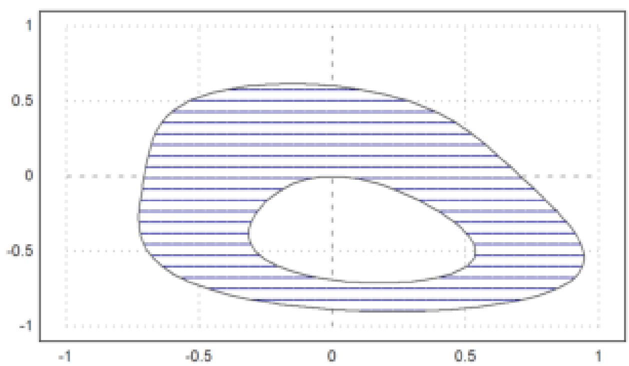
Jika x adalah vektor yang diurutkan, dan y adalah vektor interval, maka plot2d akan memplot rentang interval yang terisi di bidang, gaya isian sama dengan gaya poligon.

```
>t=-1:0.01:1; x=~t-0.01,t+0.01~; y=x^3-x;
>plot2d(t,y):
```



Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks 2xn. Baris pertama adalah batas bawah dan baris kedua berisi batas atas.

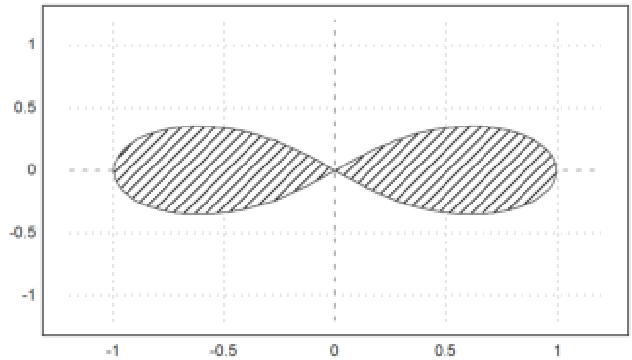
```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <= 1
```



Kita juga dapat mengisi rentang nilai seperti

$$-1 \leq (x^2 + y^2)^2 - x^2 + y^2 \leq 0.$$

```
>plot2d("(x^2+y^2)^2-x^2+y^2",r=1.2,level=[-1;0],style="/"):
```

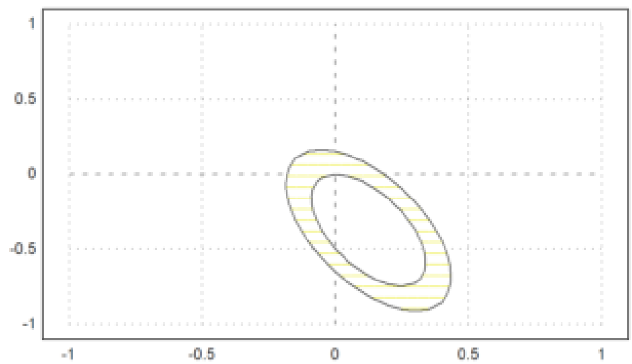


```
>plot2d("cos(x)","sin(x)^3",xmin=0,xmax=2pi,>filled,style="/"):
```

Contoh Tambahan:
Buatlah daerah yang dibatasi kurva dari fungsi tersebut.

$$3x^2 + 2yx + y^2 + \frac{y}{2}$$

```
>expr:="3*x^2+2*y*x+y^2+y*0.5";
>plot2d(expr,level=[0;0.1],style="-",color=yellow):
```



Grafik Fungsi Parametrik

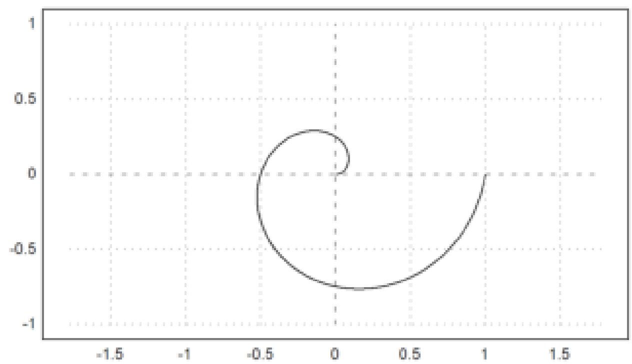
Nilai x tidak perlu diurutkan. (x,y) hanya menggambarkan sebuah kurva. Jika x diurutkan, kurva tersebut adalah grafik dari sebuah fungsi.

Dalam contoh berikut, kami memplot spiral

$$\gamma(t) = t \cdot (\cos(2\pi t), \sin(2\pi t))$$

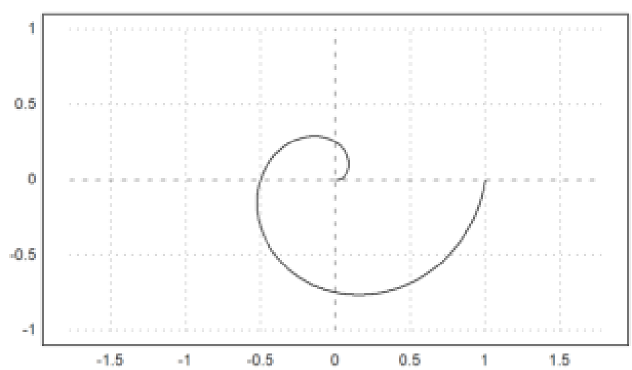
Kita perlu menggunakan sangat banyak titik untuk tampilan yang halus atau fungsi adaptive() untuk mengevaluasi ekspresi (lihat fungsi adaptive() untuk lebih jelasnya).

```
>t=linspace(0,1,1000); ...
>plot2d(t*cos(2*pi*t),t*sin(2*pi*t),r=1):
```

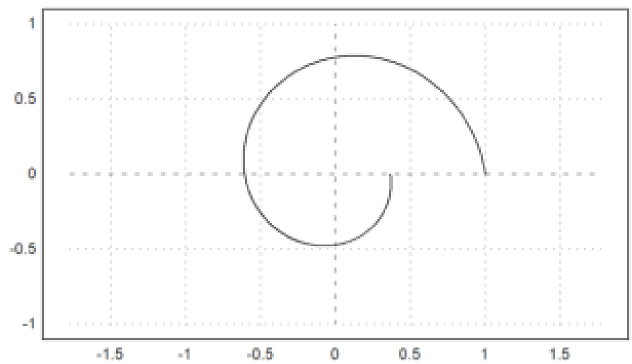


Sebagai alternatif, Anda dapat menggunakan dua ekspresi untuk kurva. Berikut ini adalah plot kurva yang sama seperti di atas.

```
>plot2d("x*cos(2*pi*x)","x*sin(2*pi*x)",xmin=0,xmax=1,r=1):
```



```
>t=linspace(0,1,1000); r=exp(-t); x=r*cos(2pi*t); y=r*sin(2pi*t);
>plot2d(x,y,r=1):
```



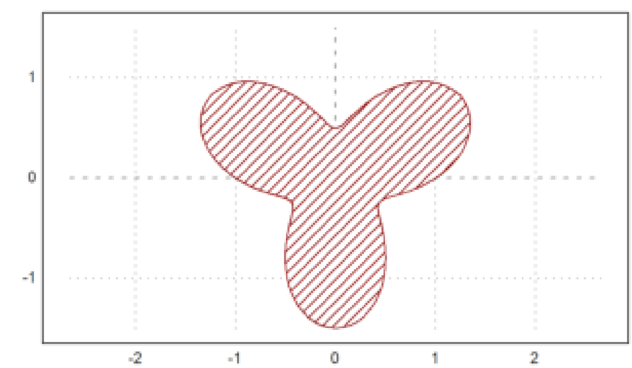
Dalam contoh berikut, kami memplot kurva

$$\gamma(t) = (r(t)\cos(t), r(t)\sin(t))$$

dengan

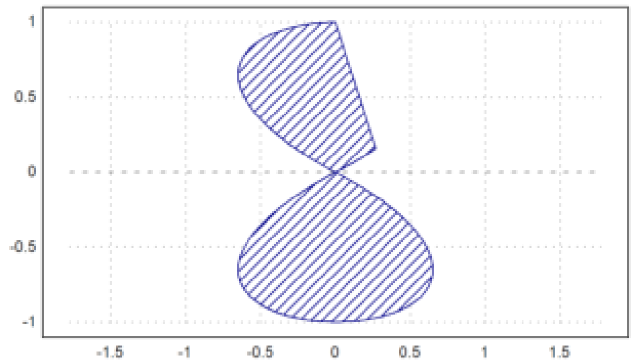
$$r(t) = 1 + \frac{\sin(3t)}{2}.$$

```
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t); ...
>plot2d(x,y,>filled,fillcolor=red,style="/",r=1.5):
```



Contoh Tambahan:
Buatlah kurva dari fungsi tersebut.

```
>t=linspace(1,2pi,1000); r=(1+cos(2*t))/2; x=r*sin(2*t); y=r*cos(t); ...
>plot2d(x,y,>filled,fillcolor=blue,style="/",r=1):
```



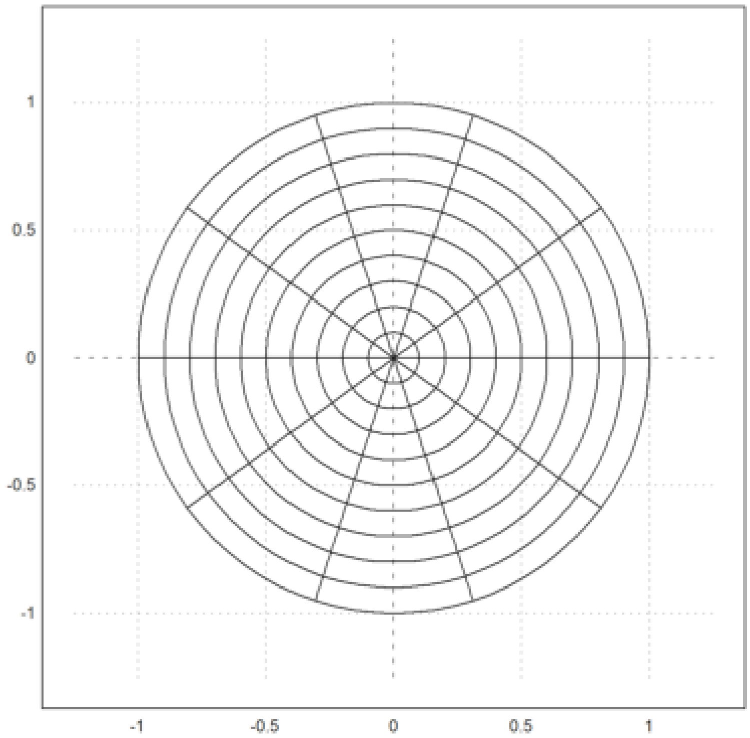
Menggambar Grafik Bilangan Kompleks

Larik bilangan kompleks juga dapat diplot. Kemudian titik-titik kisi akan dihubungkan. Jika sejumlah garis kisi ditentukan (atau vektor 1x2 garis kisi) pada argumen cgrid, hanya garis-garis kisi tersebut yang akan terlihat.

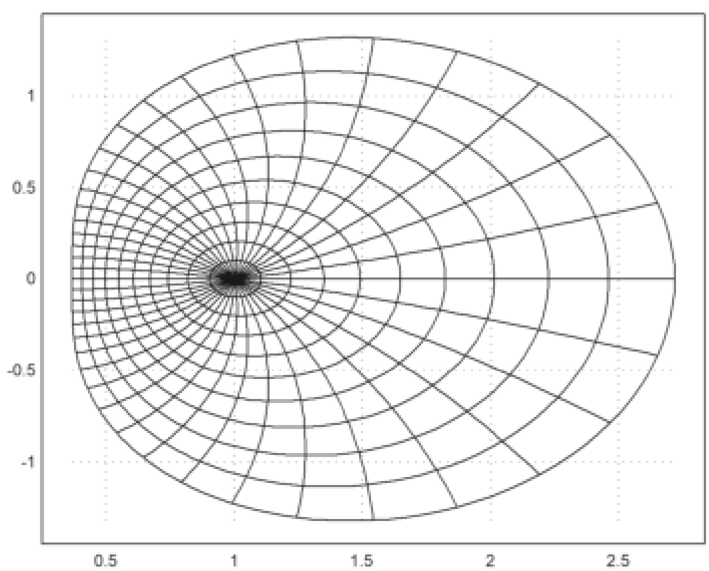
Matriks bilangan kompleks akan secara otomatis diplot sebagai kisi-kisi pada bidang kompleks.

Pada contoh berikut, kami memplot gambar lingkaran satuan di bawah fungsi eksponensial. Parameter cgrid menyembunyikan beberapa kurva kisi-kisi.

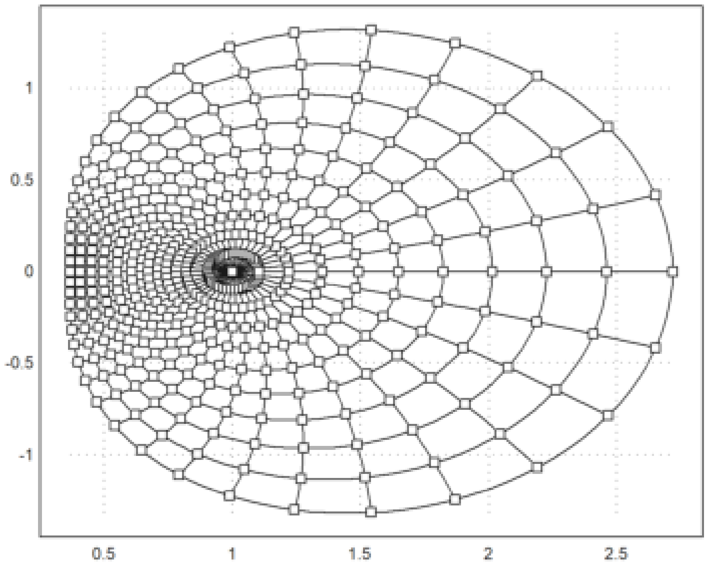
```
>aspect(); r=linspace(0,1,50); a=linspace(0,2pi,80)'; z=r*exp(I*a);...
>plot2d(z,a=-1.25,b=1.25,c=-1.25,d=1.25,cgrid=10):
```




```
>aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)'; z=r*exp(I*a);
>plot2d(exp(z),cgrid=[40,10]):
```




```
>r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=r*exp(I*a);
>plot2d(exp(z),>points,>add):
```

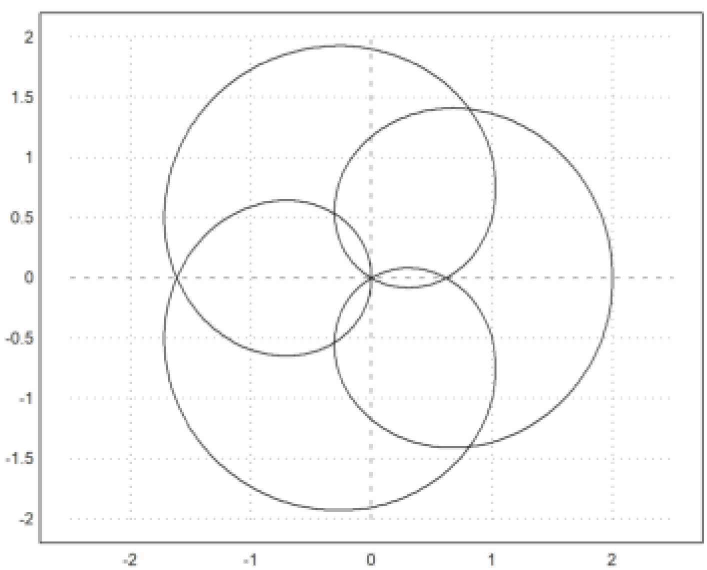


Vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian riil dan bagian imajiner.

Dalam contoh, kami memplot lingkaran satuan dengan

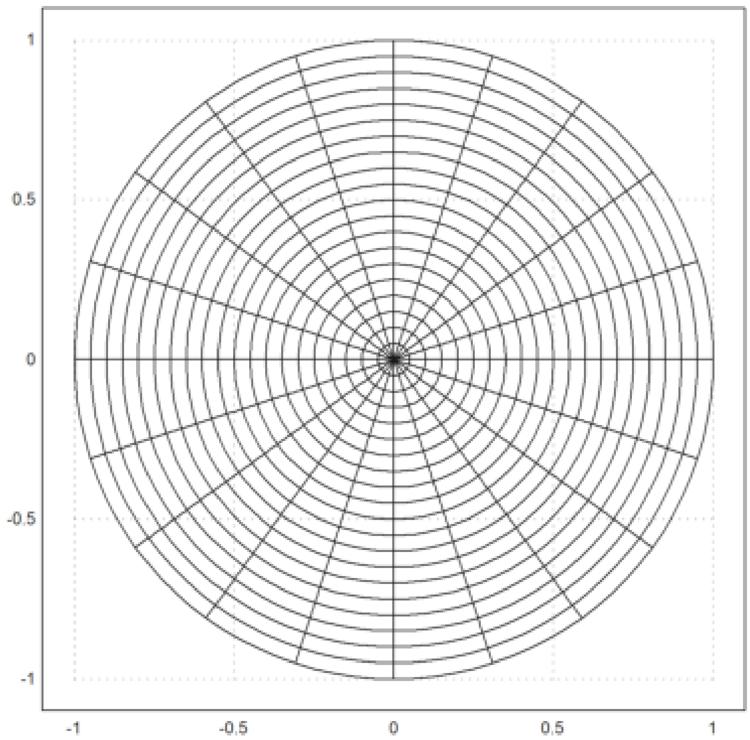
$$\gamma(t) = e^{it}$$

```
>t=linspace(0,2pi,1000); ...
>plot2d(exp(I*t)+exp(4*I*t),r=2):
```



Contoh Tambahan:

```
>aspect(); r=linspace(0,1,100); a=linspace(0,2pi,100)'; z=r*exp(I*a); ...
>plot2d(z,a=-1,b=1,c=-1,d=1,cgrid=20):
```

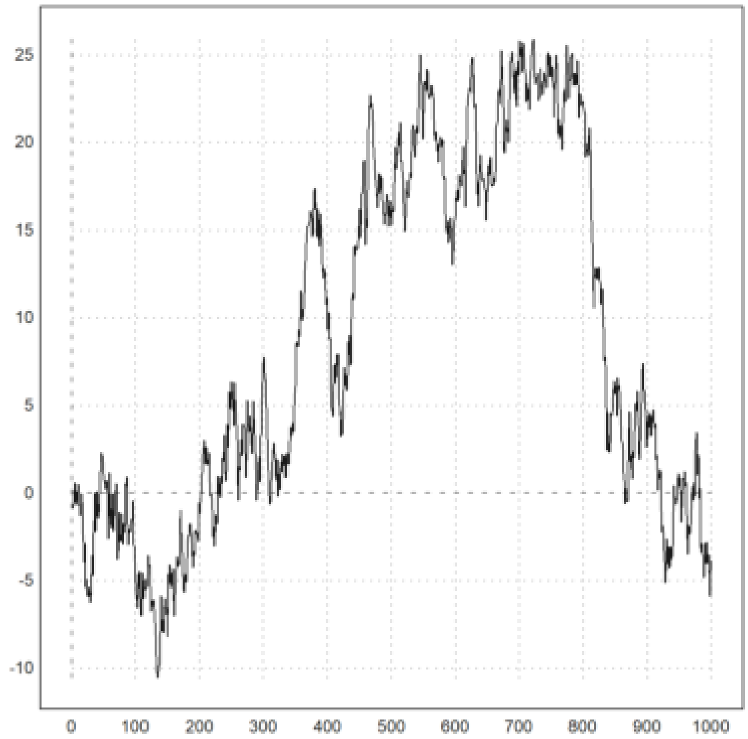


Plot Statistik

Terdapat banyak fungsi yang dikhususkan pada plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

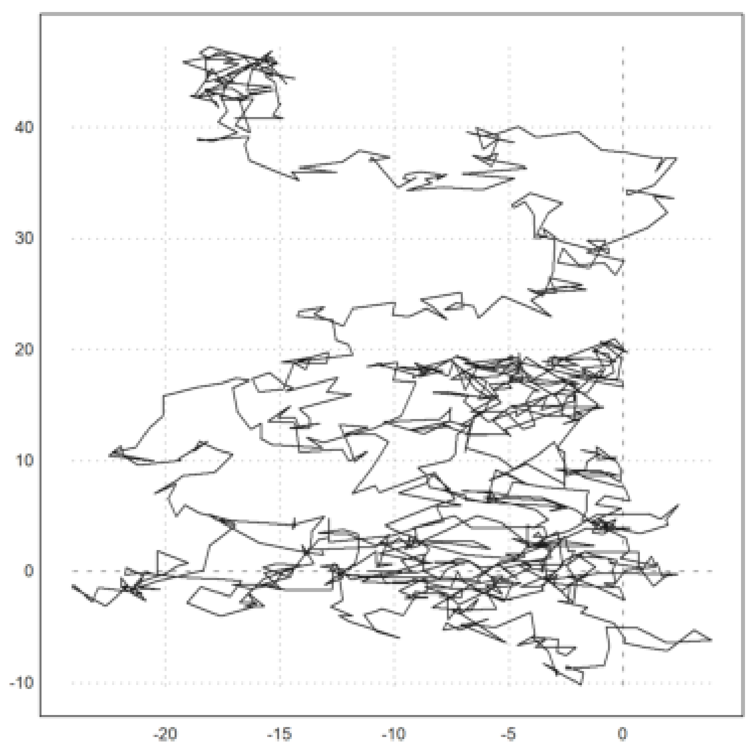
Jumlah kumulatif dari nilai berdistribusi normal 0-1 menghasilkan jalan acak.

```
>plot2d(cumsum(randnormal(1,1000))):
```

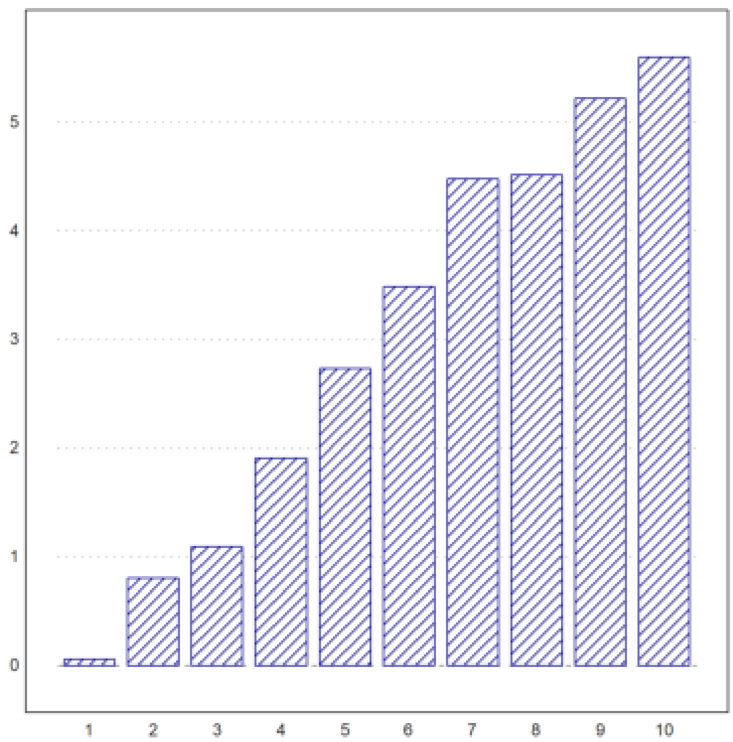


Dengan menggunakan dua baris, ini menunjukkan jalan kaki dalam dua dimensi.

```
>X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]):
```

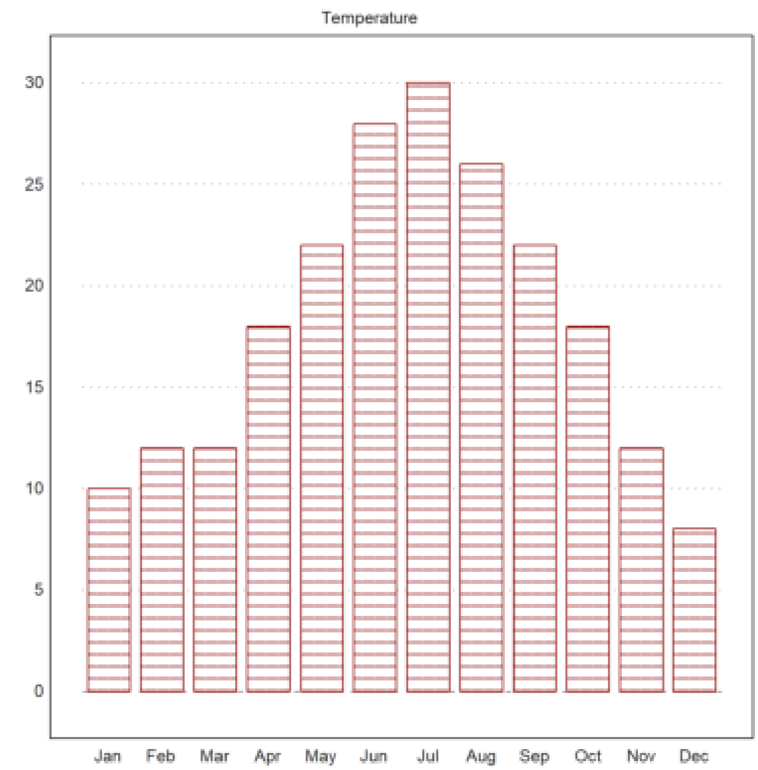


```
>columnsplot(cumsum(random(10)),style="/",color=blue):
```

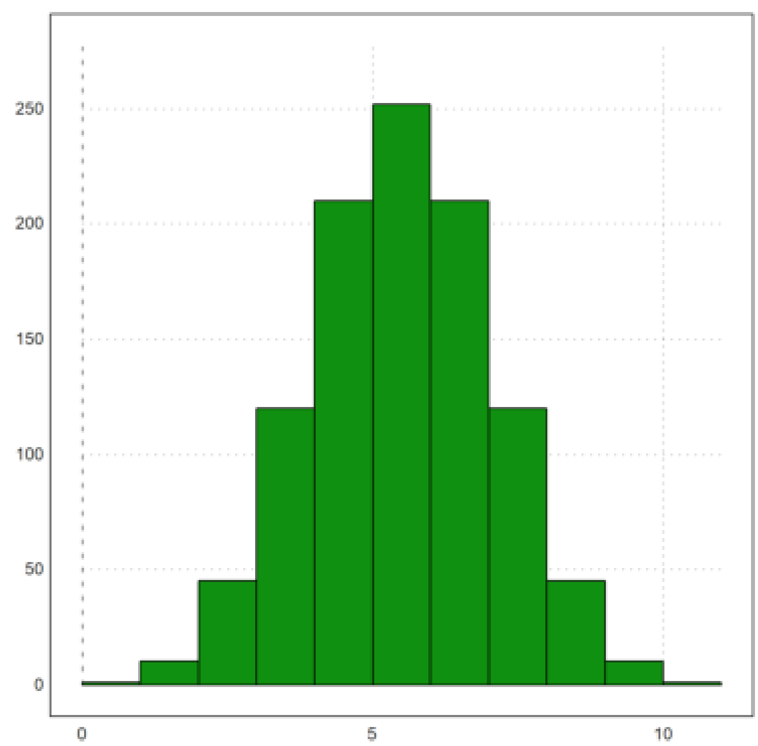


Ini juga dapat menampilkan string sebagai label.

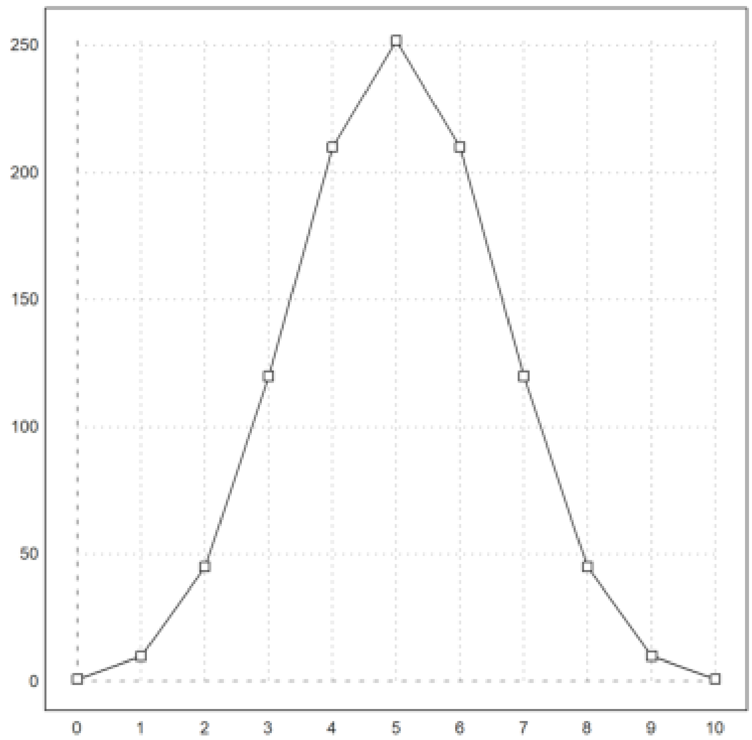
```
>months=["Jan","Feb","Mar","Apr","May","Jun", ...
>  "Jul","Aug","Sep","Oct","Nov","Dec"];
>values=[10,12,12,18,22,28,30,26,22,18,12,8];
>columnsplot(values,lab=months,color=red,style="-");
>title("Temperature"):
```



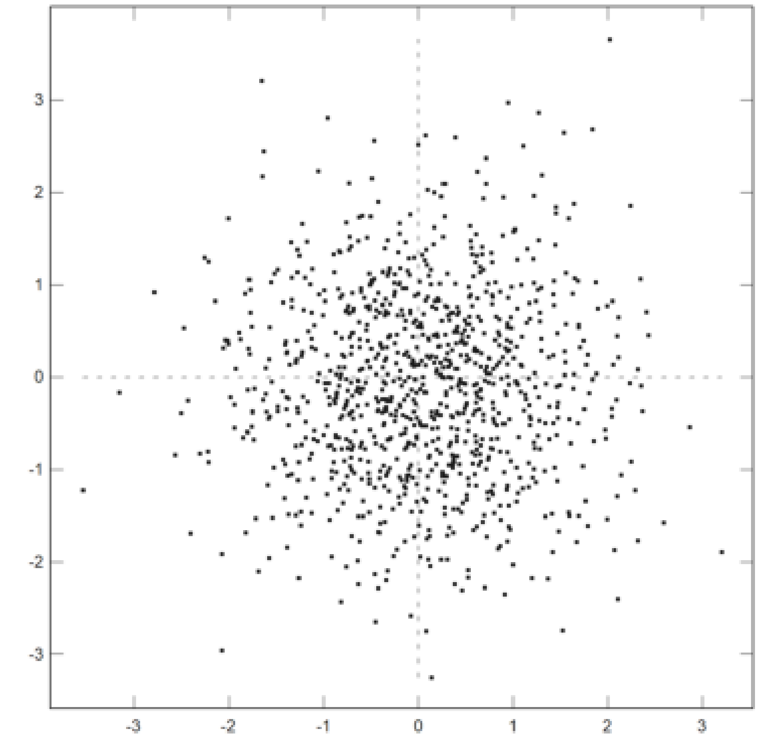

```
>k=0:10;
>plot2d(k,bin(10,k),>bar):
```



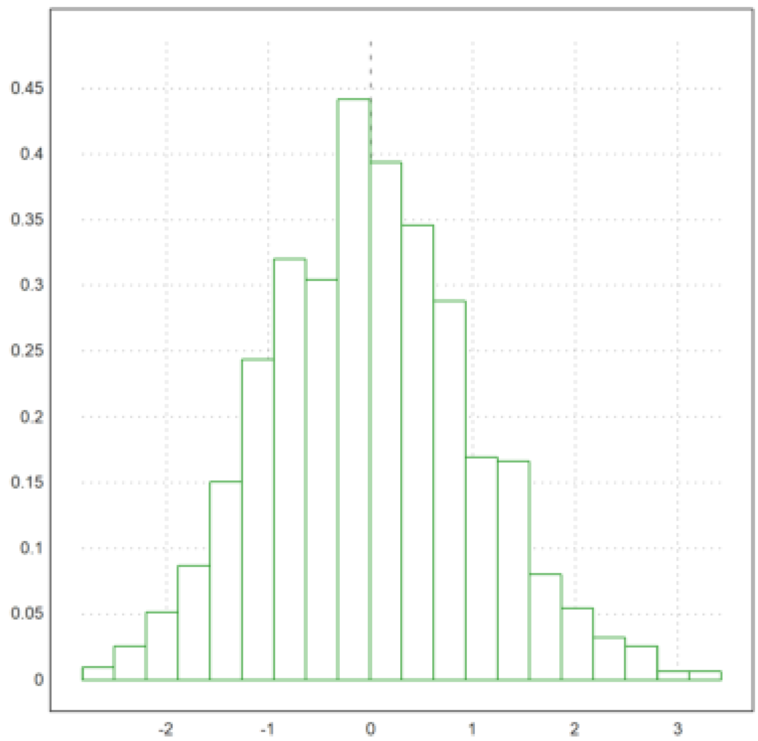
```
>plot2d(k,bin(10,k)); plot2d(k,bin(10,k),>points,>add):
```



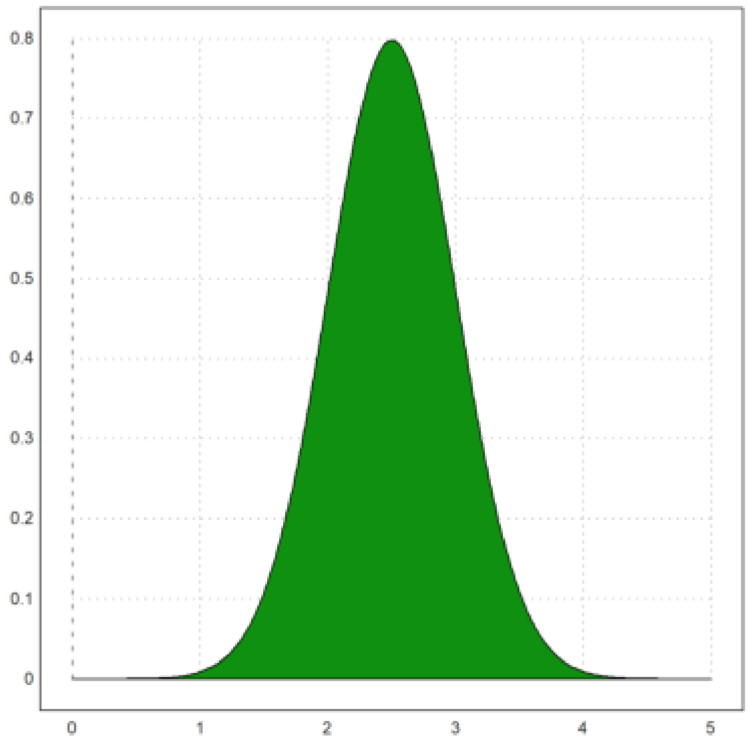

```
>plot2d(normal(1000),normal(1000),>points,grid=6,style=".."):
```




```
>plot2d(normal(1,1000),>distribution,style="O"):
```

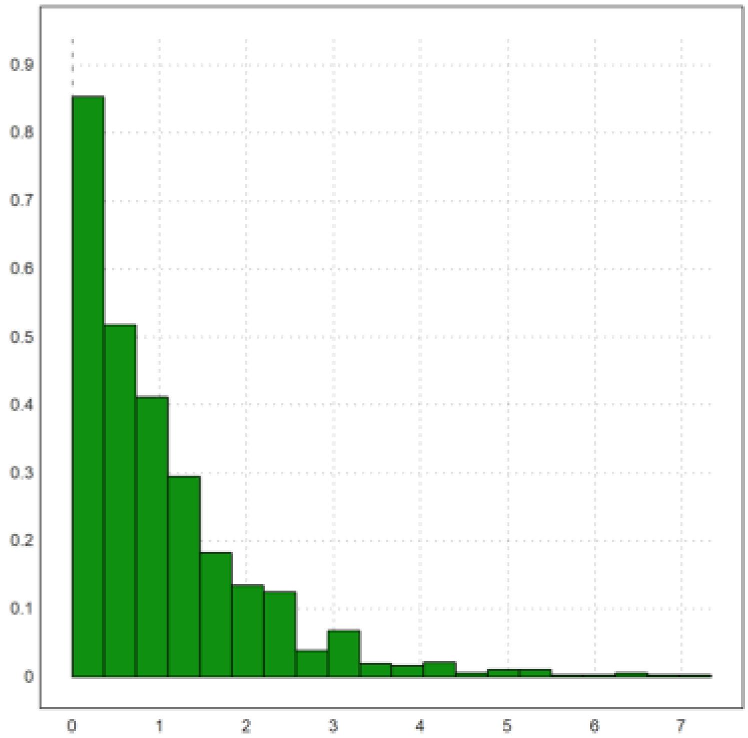


```
>plot2d("qnormal",0,5;2.5,0.5,>filled):
```



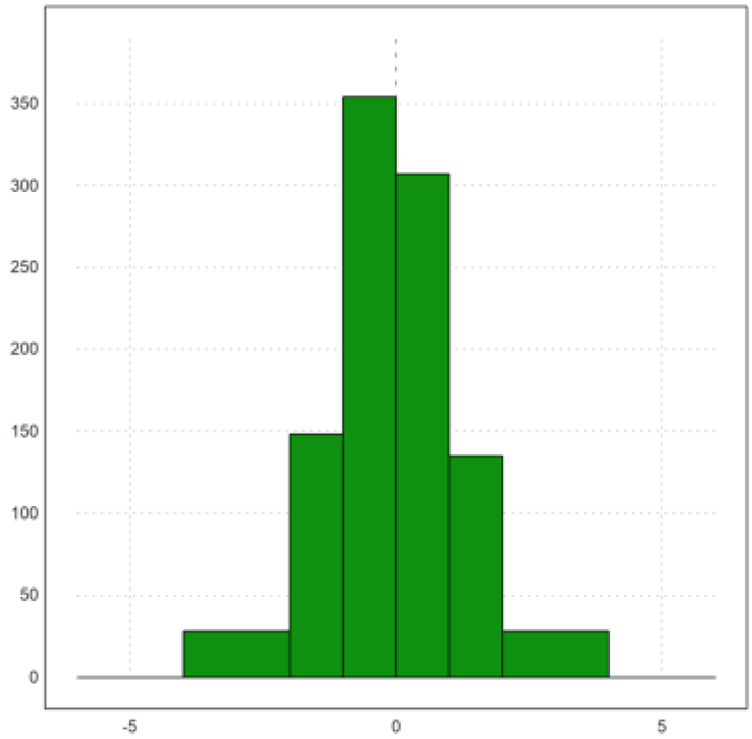
Untuk memplot distribusi statistik eksperimental, Anda dapat menggunakan distribution=n dengan plot2d.

```
>w=randexponential(1,1000); // exponential distribution
>plot2d(w,>distribution): // or distribution=n with n intervals
```



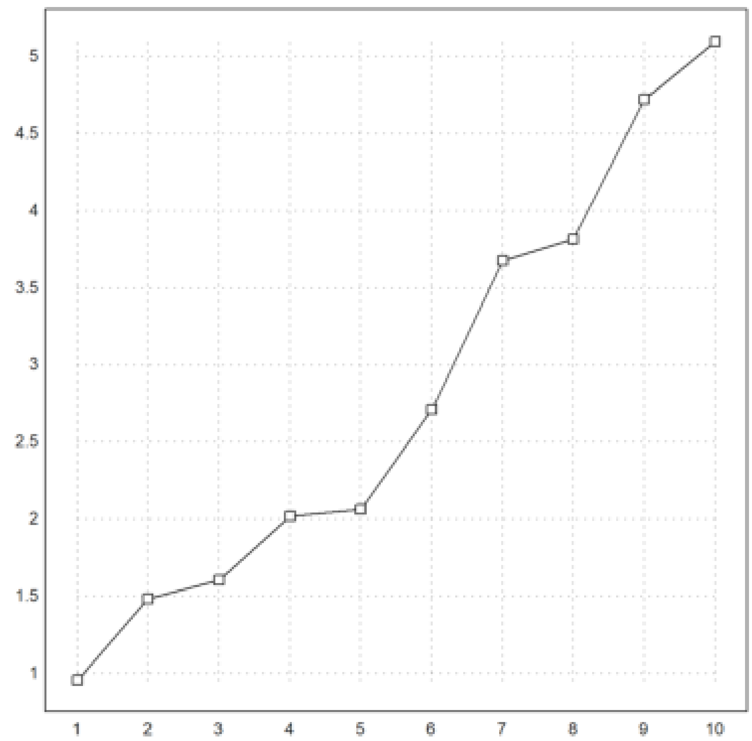
Atau Anda dapat menghitung distribusi dari data dan memplot hasilnya dengan >bar di plot3d, atau dengan plot kolom.

```
>w=normal(1000); // 0-1-normal distribution
>{x,y}=histo(w,10,v=[-6,-4,-2,-1,0,1,2,4,6]); // interval bounds v
>plot2d(x,y,>bar):
```

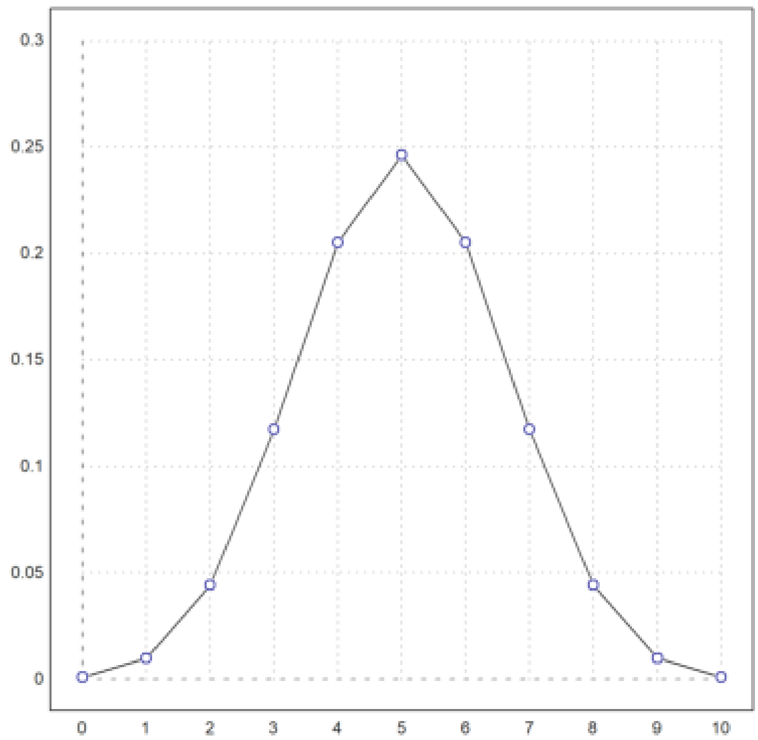


Fungsi statplot() menetapkan gaya dengan string sederhana.

```
>statplot(1:10,cumsum(random(10)),"b"):
```



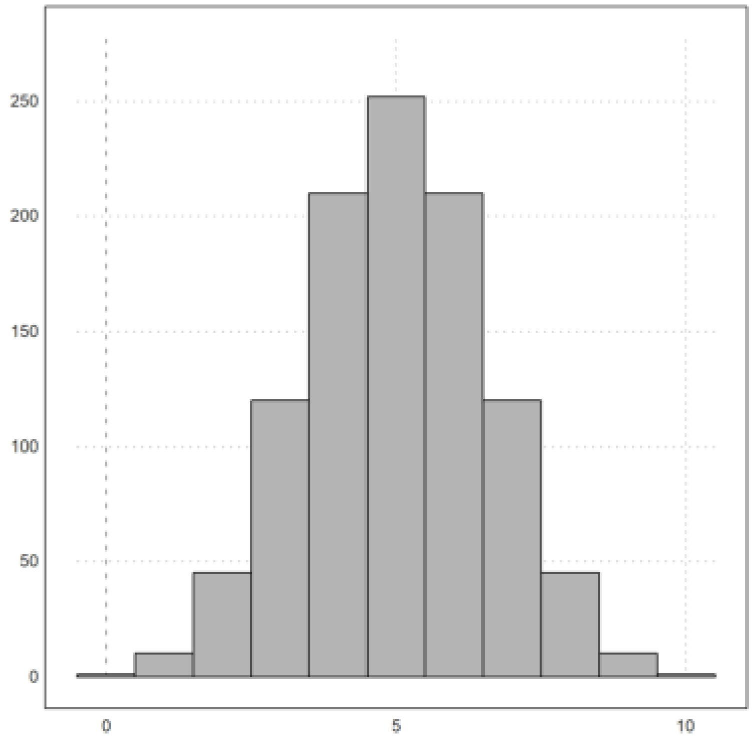
```
>n=10; i=0:n; ...
>plot2d(i,bin(n,i)/2^n,a=0,b=10,c=0,d=0.3); ...
>plot2d(i,bin(n,i)/2^n,points=true,style="ow",add=true,color=blue):
```



Selain itu, data dapat diplot sebagai batang. Dalam hal ini, x harus diurutkan dan satu elemen lebih panjang dari y. Batang akan memanjang dari x[i] ke x[i+1] dengan nilai y[i]. Jika x memiliki ukuran yang sama dengan y, maka x akan diperpanjang satu elemen dengan jarak terakhir.

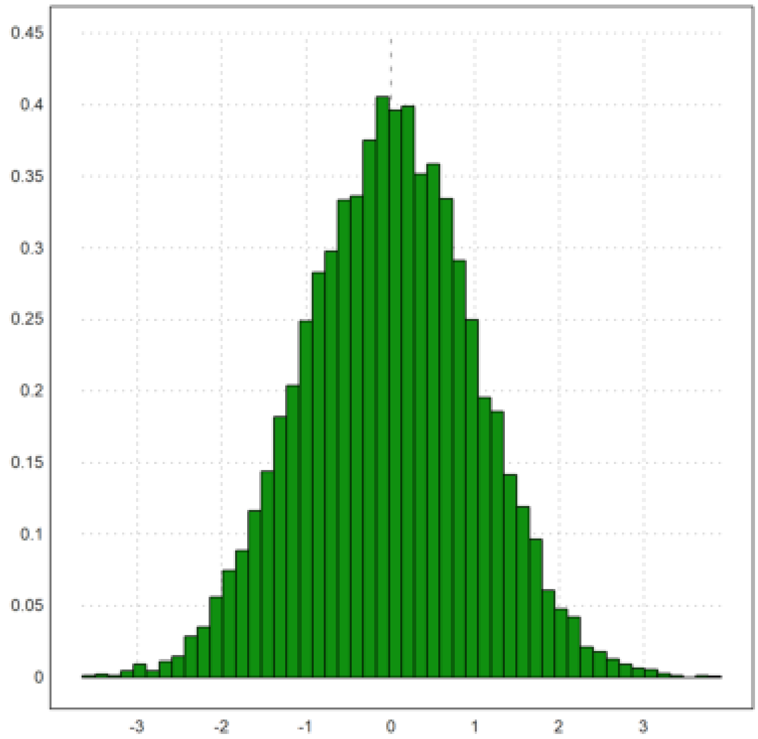
Gaya isian dapat digunakan seperti di atas.

```
>n=10; k=bin(n,0:n); ...
>plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray):
```

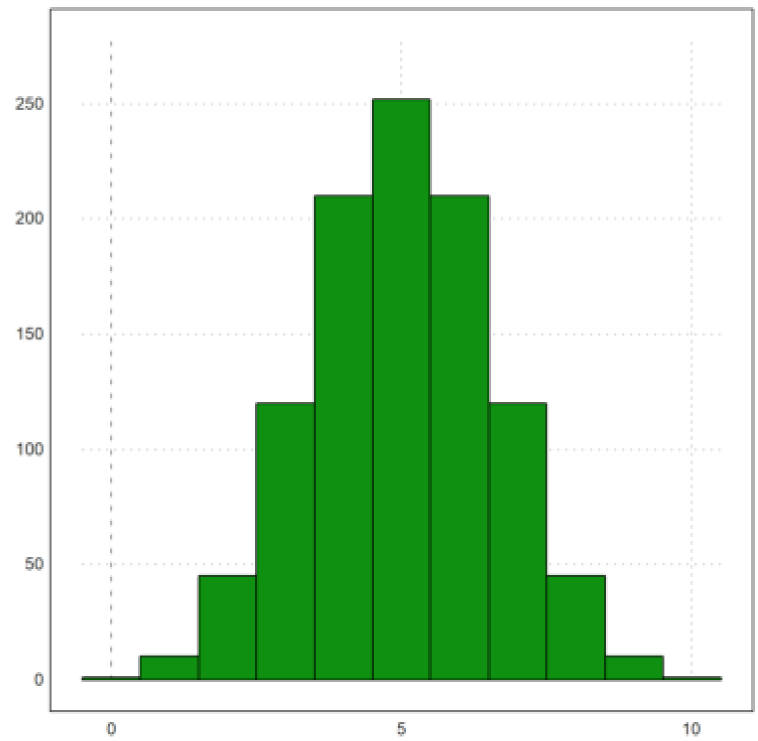


Data untuk plot batang (batang = 1) dan histogram (histogram = 1) dapat diberikan secara eksplisit dalam xv dan yv, atau dapat dihitung dari distribusi empiris dalam xv dengan >distribusi (atau distribusi = n). Histogram dari nilai xv akan dihitung secara otomatis dengan >histogram. Jika >even ditentukan, nilai xv akan dihitung dalam interval bilangan bulat.

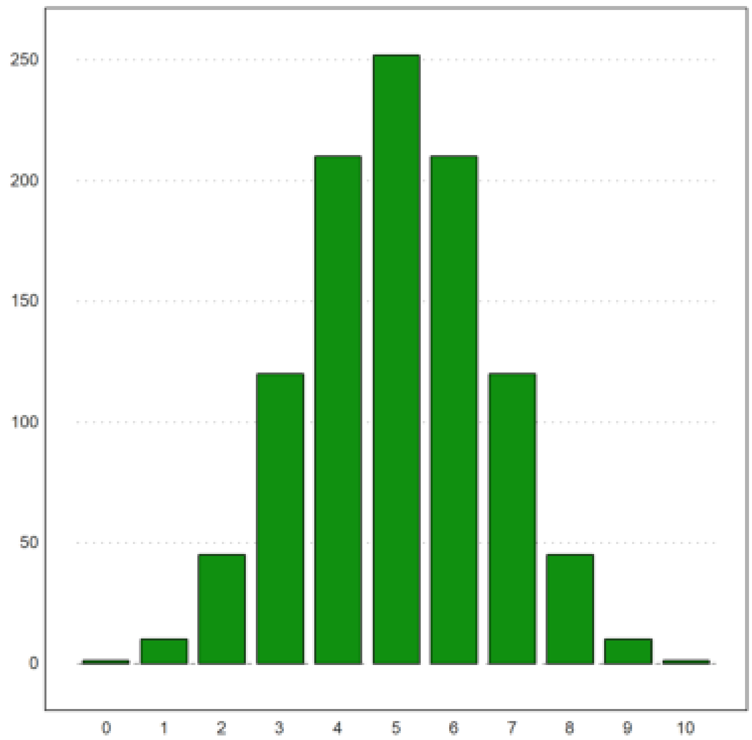
```
>plot2d(normal(10000),distribution=50):
```



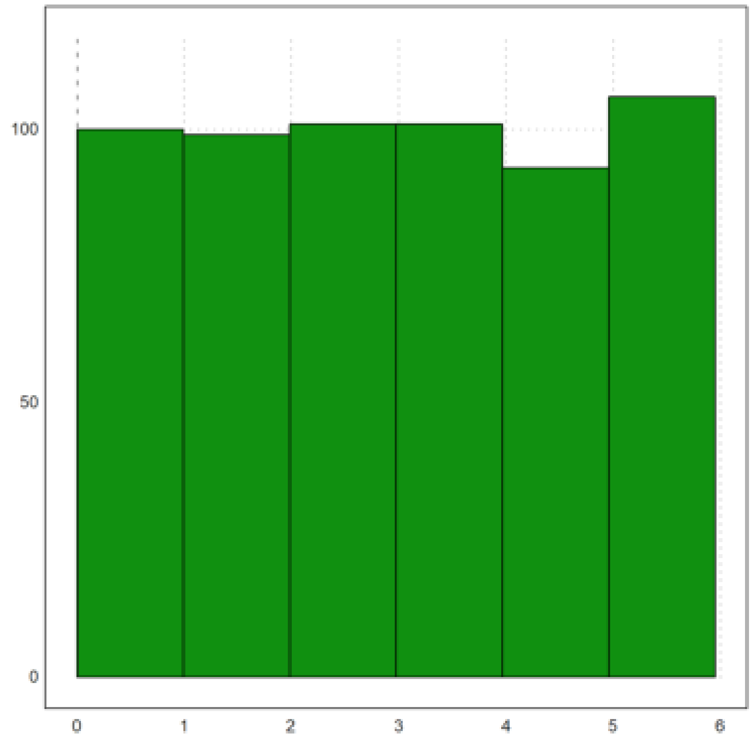

```
>k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m,>bar):
```



```
>columnsplot(m,k):
```

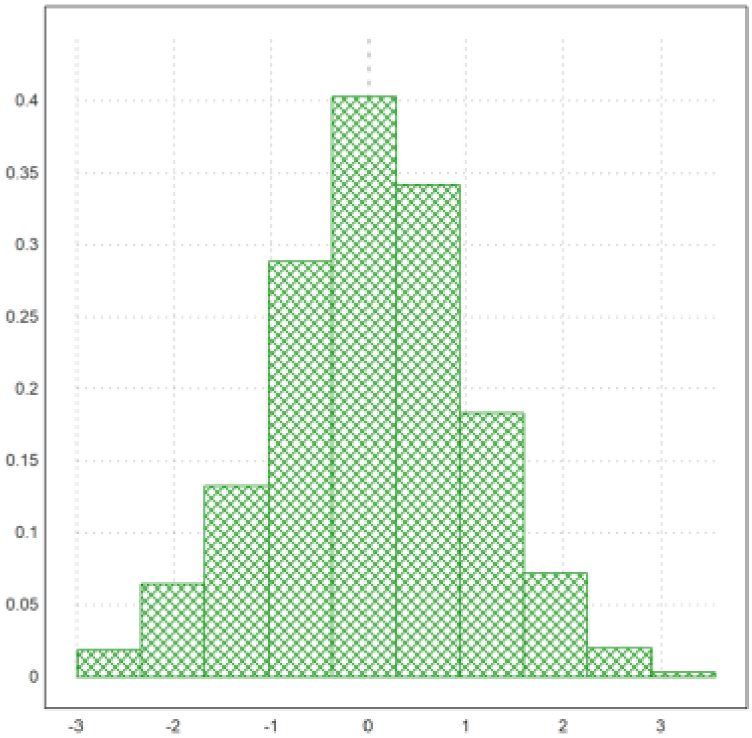



```
>plot2d(random(600)*6,histogram=6):
```



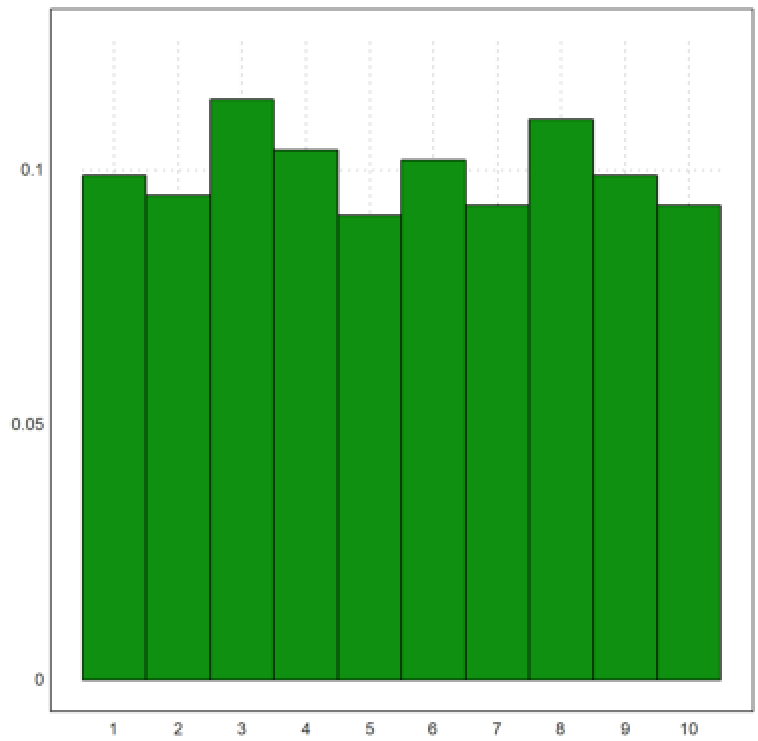
Untuk distribusi, ada parameter distribution=n, yang menghitung nilai secara otomatis dan mencetak distribusi relatif dengan n sub-interval.

```
>plot2d(normal(1,1000),distribution=10,style="\/"):
```



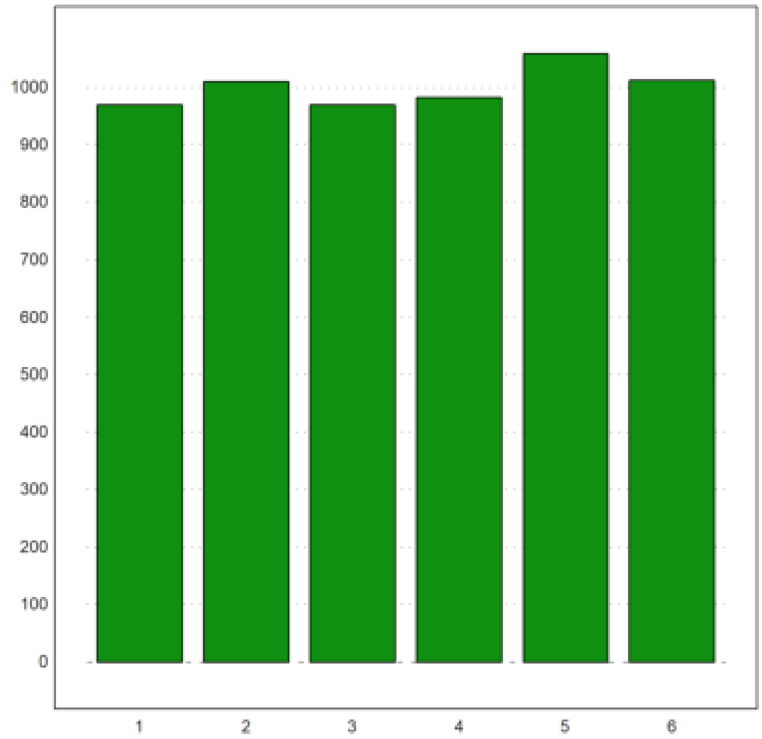
Dengan parameter even=true, ini akan menggunakan interval bilangan bulat.

```
>plot2d(intrandom(1,1000,10),distribution=10,even=true):
```

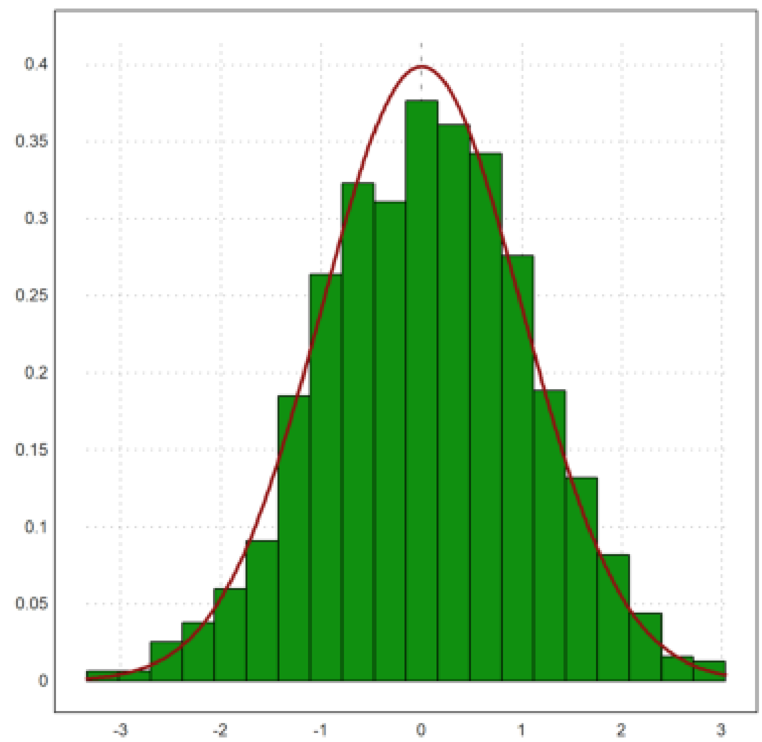


Perhatikan bahwa ada banyak plot statistik yang mungkin berguna. Lihatlah tutorial tentang statistik.

```
>columnsplot(getmultiplicities(1:6,intrandom(1,6000,6))):
```

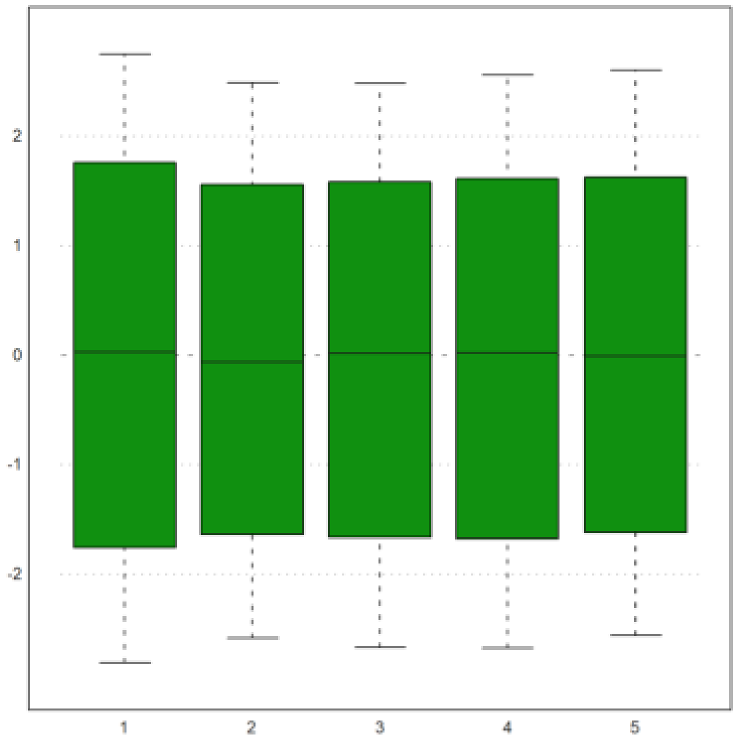



```
>plot2d(normal(1,1000),>distribution); ...
>  plot2d("qnormal(x)",color=red,thickness=2,>add):
```



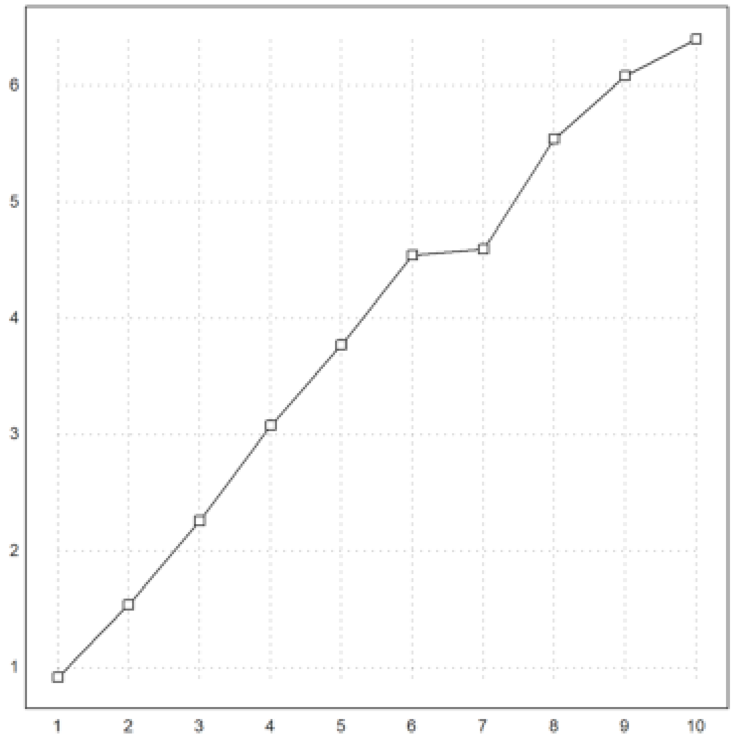
Ada juga banyak plot khusus untuk statistik. Boxplot menunjukkan kuartil dari distribusi ini dan banyak pencilan. Menurut definisi, pencilan dalam boxplot adalah data yang melebihi 1,5 kali kisaran 50% tengah plot.

```
>M=normal(5,1000); boxplot(quartiles(M)):
```



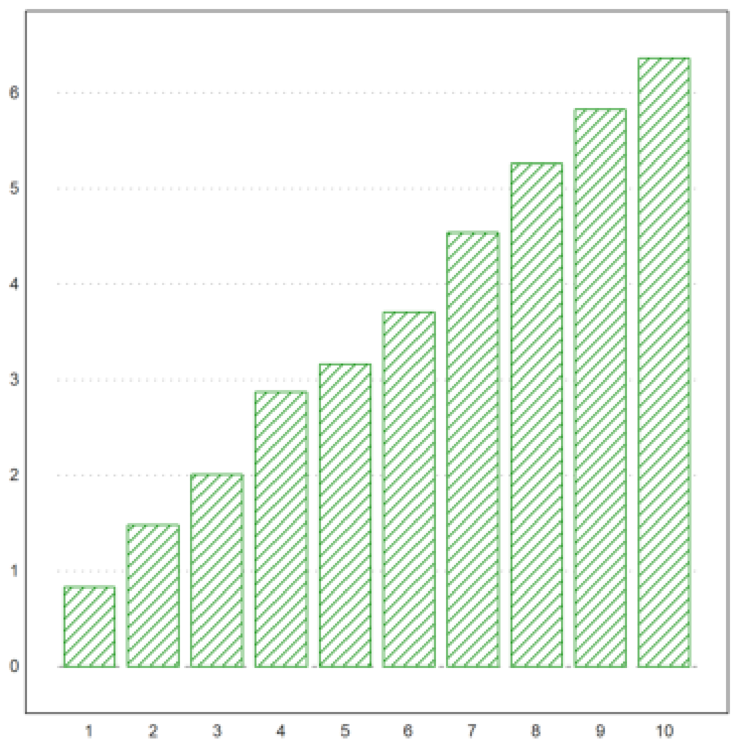
Contoh Tambahan:
Buatlah plot dalam bentuk string sederhana!

```
>statplot(1:10,cumsum(random(10)),"b"):
```



2. Tentukan plot dalam bentuk histogram sebanyak 10!

```
>columnsplot(cumsum(random(10)),style="/",color=green):
```



Plot implisit menunjukkan garis level yang menyelesaikan f(x,y)=level, di mana "level" dapat berupa nilai tunggal atau vektor nilai. Jika level = "auto", akan ada nc garis level, yang akan menyebar di antara nilai minimum dan maksimum fungsi secara merata. Warna yang lebih gelap atau lebih terang dapat ditambahkan dengan >hue untuk mengindikasikan nilai fungsi. Untuk fungsi implisit, xv haruslah sebuah fungsi atau ekspresi dari parameter x dan y, atau, sebagai alternatif, xv dapat berupa sebuah matriks nilai.

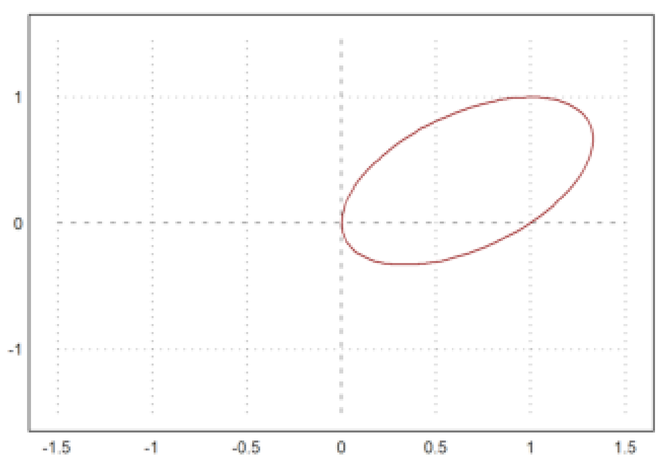
Euler dapat menandai garis level

$$f(x, y) = c$$

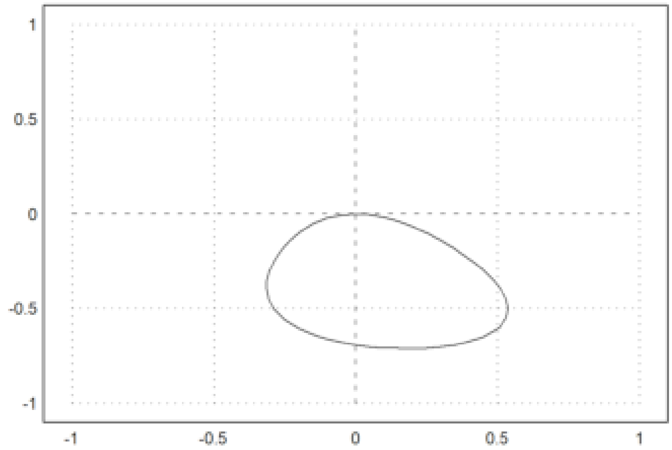
dari fungsi apa pun.

Untuk menggambar himpunan f(x,y)=c untuk satu atau lebih konstanta c, Anda dapat menggunakan plot2d() dengan plot implisitnya pada bidang. Parameter untuk c adalah level = c, di mana c dapat berupa vektor garis level. Sebagai tambahan, sebuah skema warna dapat digambar pada latar belakang untuk mengindikasikan nilai fungsi untuk setiap titik pada plot. Parameter "n" menentukan kehalusan plot.

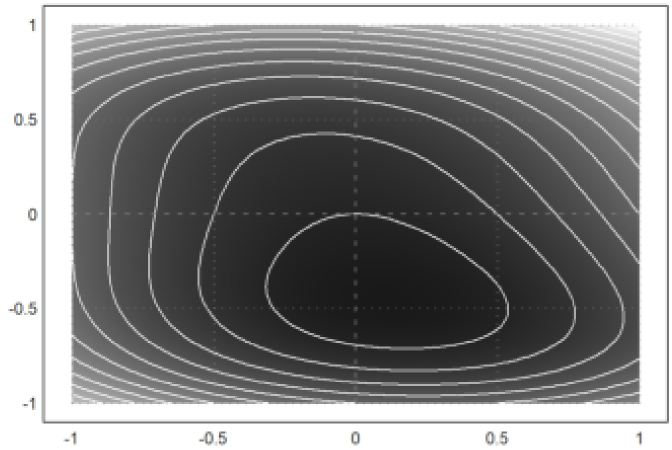
```
>aspect(1.5);
>plot2d("x^2+y^2-x*y-x",r=1.5,level=0,contourcolor=red):
```



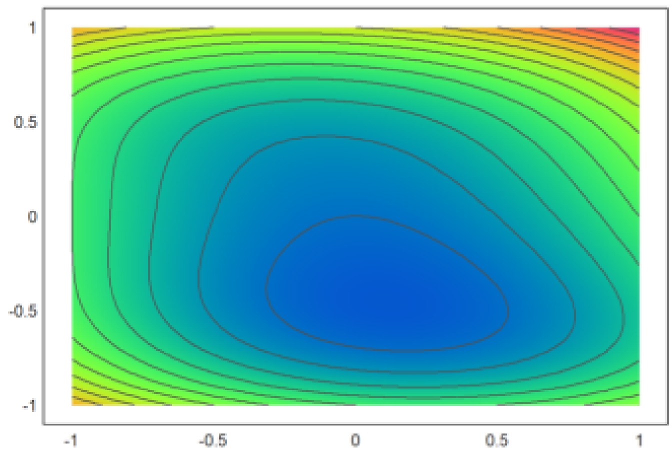

```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr,level=0): // Solutions of f(x,y)=0
```




```
>plot2d(expr,level=0:0.5:20,>hue,contourcolor=white,n=200): // nice
```

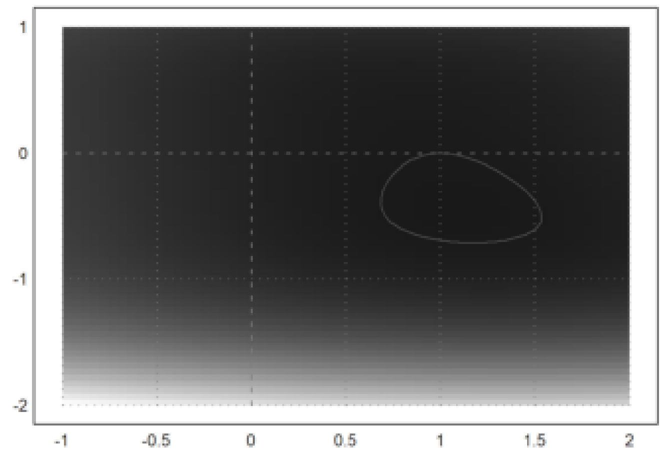



```
>plot2d(expr,level=0:0.5:20,>hue,>spectral,n=200,grid=4): // nicer
```

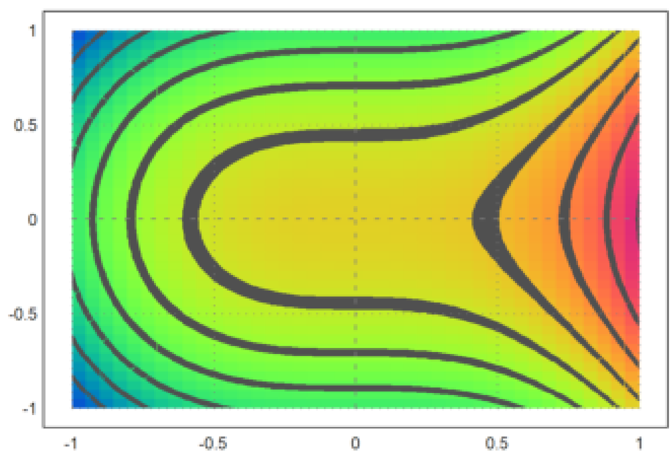


Hal ini juga berlaku untuk plot data. Tetapi Anda harus menentukan rentang untuk label sumbu.

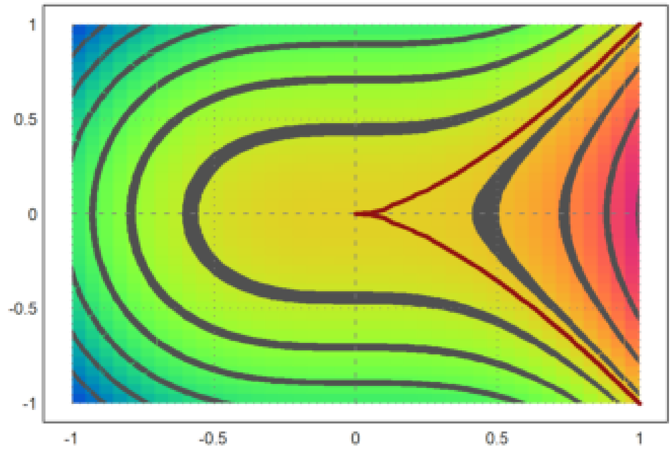
```
>x=-2:0.05:1; y=x'; z=expr(x,y);
>plot2d(z,level=0,a=-1,b=2,c=-2,d=1,>hue):
```



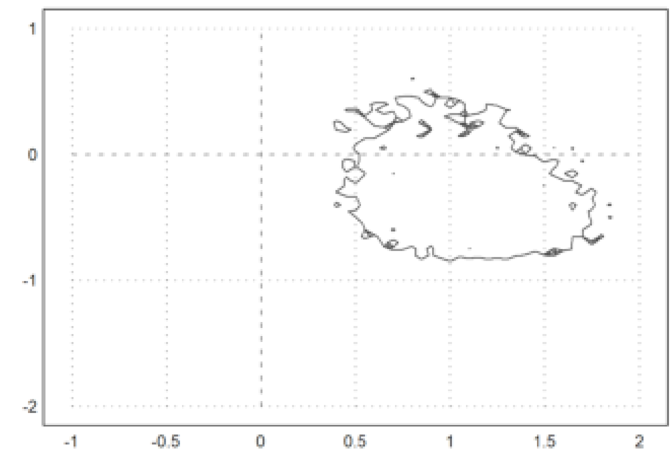
```
>plot2d("x^3-y^2",>contour,>hue,>spectral):
```



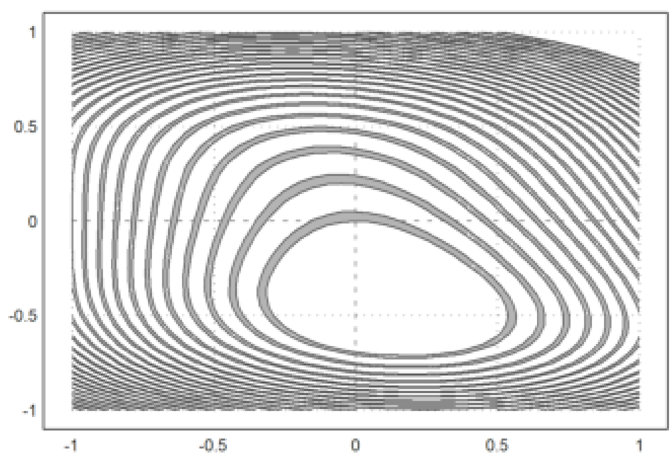
```
>plot2d("x^3-y^2",level=0,contourwidth=3,>add,contourcolor=red):
```



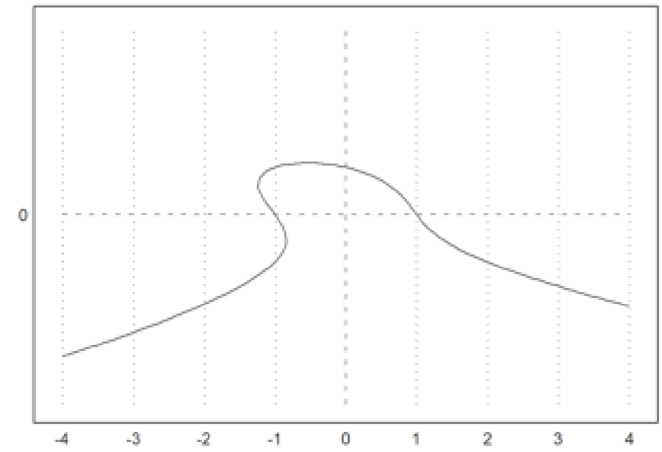

```
>z=z+normal(size(z))*0.2;
>plot2d(z,level=0.5,a=-1,b=2,c=-2,d=1):
```



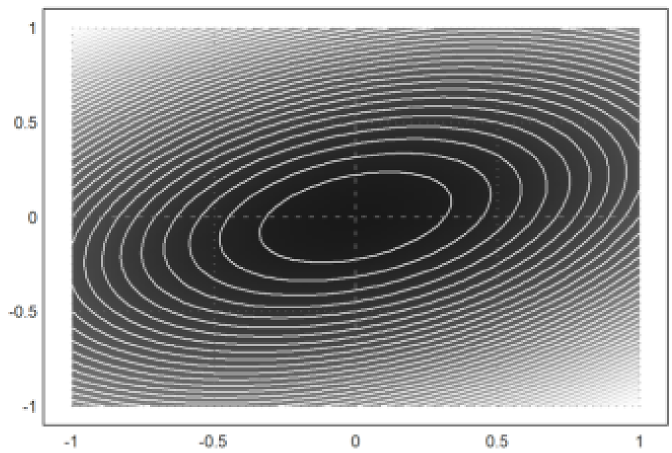

```
>plot2d(expr,level=[0:0.2:5;0.05:0.2:5.05],color=lightgray):
```



```
>plot2d("x^2+y^3+x*y",level=1,r=4,n=100):
```




```
>plot2d("x^2+2*y^2-x*y",level=0:0.1:10,n=100,contourcolor=white,>hue):
```



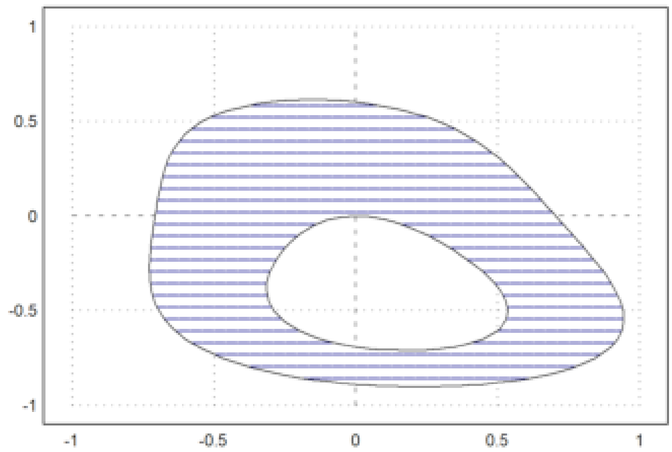
Dimungkinkan juga untuk mengisi set

$$a \le f(x,y) \le b$$

dengan rentang level.

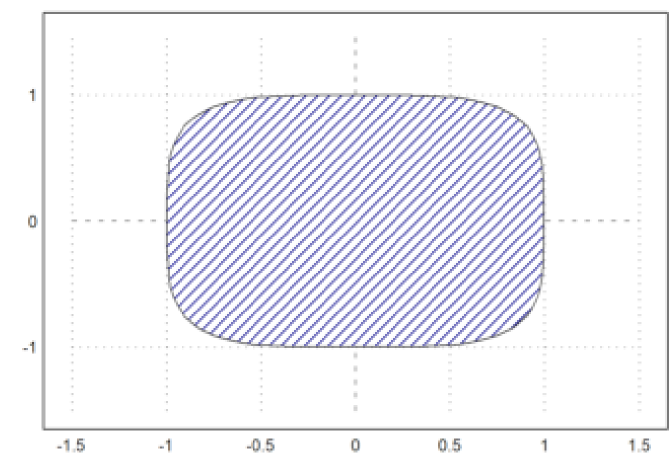
Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks 2xn. Baris pertama adalah batas bawah dan baris kedua berisi batas atas.

```
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <= 1
```

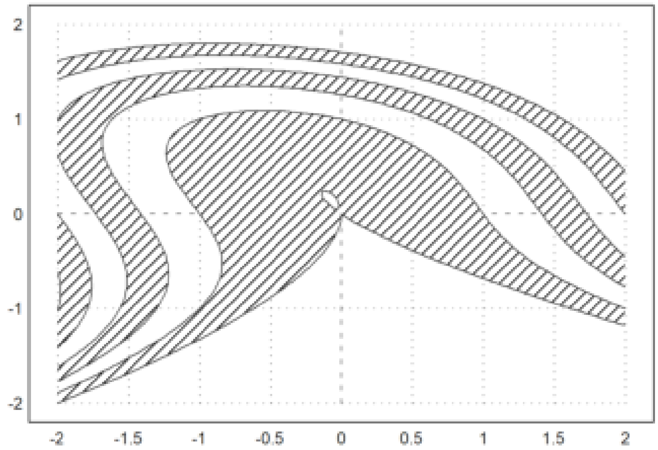


Plot implisit juga dapat menunjukkan rentang level. Maka level harus berupa matriks 2xn interval level, di mana baris pertama berisi awal dan baris kedua adalah akhir dari setiap interval. Sebagai alternatif, vektor baris sederhana dapat digunakan untuk level, dan parameter dl memperluas nilai level ke interval.

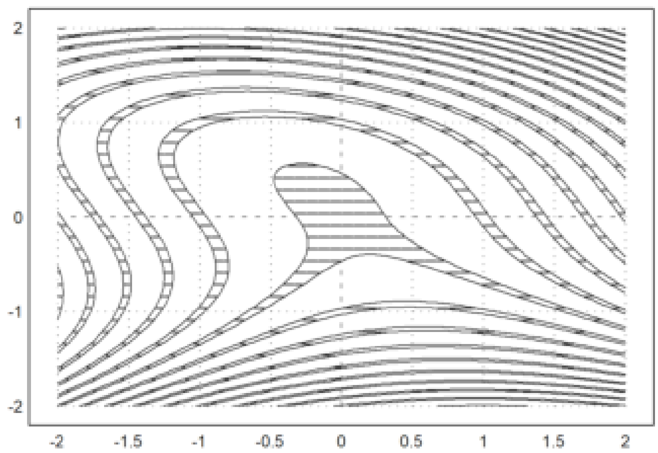
```
>plot2d("x^4+y^4",r=1.5,level=[0;1],color=blue,style="/"):
```



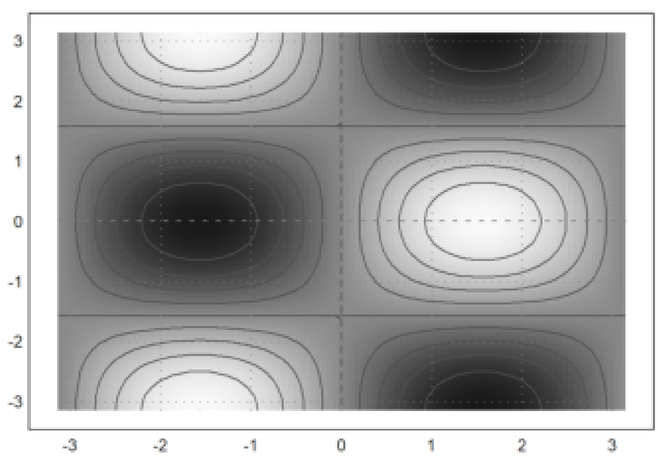
```
>plot2d("x^2+y^3+x*y",level=[0,2,4;1,3,5],style="/",r=2,n=100):
```




```
>plot2d("x^2+y^3+x*y",level=-10:20,r=2,style="-",dl=0.1,n=100):
```




```
>plot2d("sin(x)*cos(y)",r=pi,>hue,>levels,n=100):
```

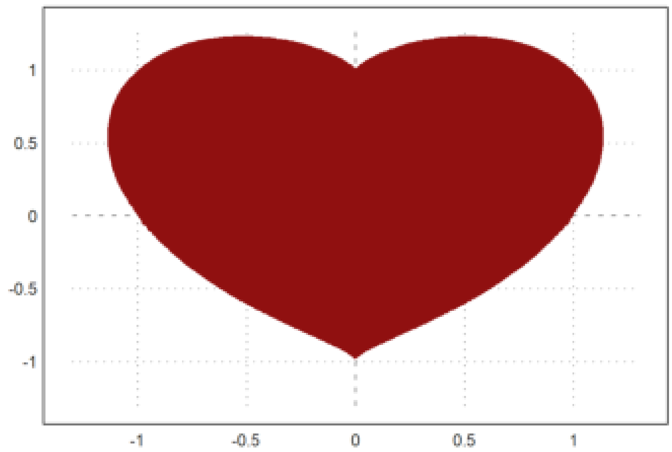


Anda juga dapat menandai suatu wilayah

$$a \leq f(x, y) \leq b.$$

Hal ini dilakukan dengan menambahkan level dengan dua baris.

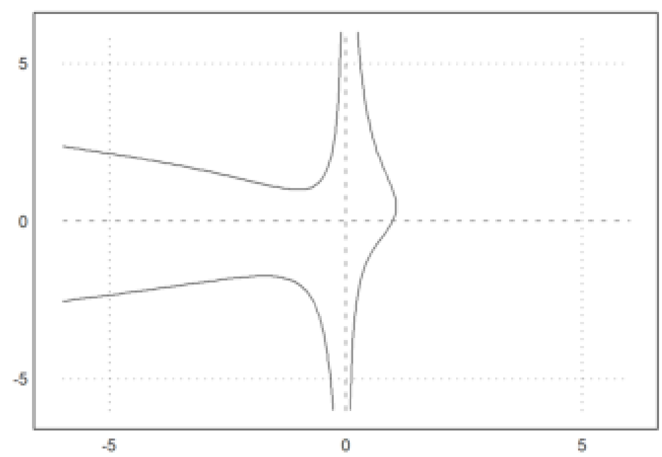
```
>plot2d("(x^2+y^2-1)^3-x^2*y^3",r=1.3, ...
>   style="#",color=red,<outline, ...
>   level=[-2;0],n=100):
```



Dimungkinkan untuk menentukan level tertentu. Misalnya, kita dapat memplot solusi dari persamaan seperti

$$x^3 - xy + x^2y^2 = 6$$

```
>plot2d("x^3-x*y+x^2*y^2",r=6,level=1,n=100):
```



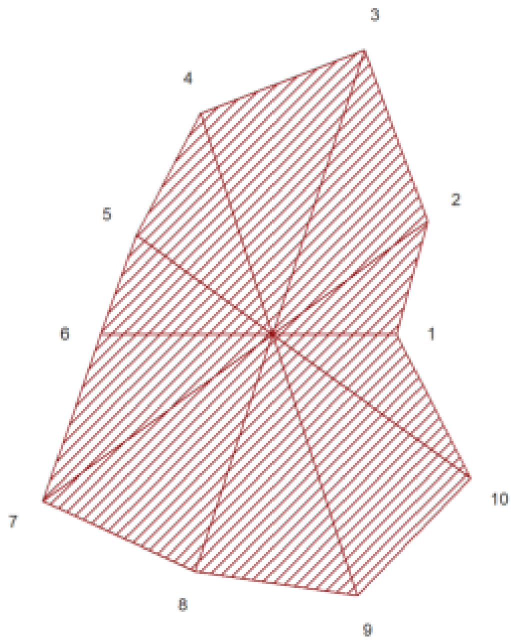
```
>function starplot1 (v, style="/", color=green, lab=none) ...
```

```
  if !holding() then clg; endif;
  w=window(); window(0,0,1024,1024);
  h=holding(1);
  r=max(abs(v))*1.2;
  setplot(-r,r,-r,r);
  n=cols(v); t=linspace(0,2pi,n);
  v=v|v[1]; c=v*cos(t); s=v*sin(t);
  cl=barcolor(color); st=barstyle(style);
  loop 1 to n
    polygon([0,c[#],c[#+1]],[0,s[#],s[#+1]],1);
    if lab!=none then
      rlab=v[#]+r*0.1;
      {col,row}=toscreen(cos(t[#])*rlab,sin(t[#])*rlab);
      ctext(""+lab[#],col,row-textheight()/2);
    endif;
  end;
  barcolor(cl); barstyle(st);
  holding(h);
  window(w);
endfunction
```

Tidak ada kisi-kisi atau kutu sumbu di sini. Selain itu, kami menggunakan jendela penuh untuk plot.

Kami memanggil reset sebelum kami menguji plot ini untuk mengembalikan default grafis. Hal ini tidak perlu dilakukan, jika Anda yakin bahwa plot Anda berfungsi.

```
>reset; starplot1(normal(1,10)+5,color=red,lab=1:10):
```



Terkadang, Anda mungkin ingin merencanakan sesuatu yang tidak dapat dilakukan oleh plot2d, tetapi hampir.

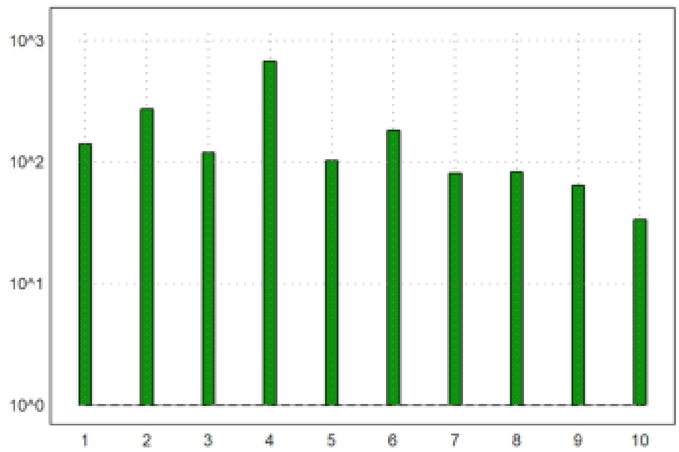
Pada fungsi berikut, kita melakukan plot impuls logaritmik. plot2d dapat melakukan plot logaritmik, tetapi tidak untuk batang impuls.

```
>function logimpulseplot1 (x,y) ...
```

```
    {x0,y0}=makeimpulse(x,log(y)/log(10));
    plot2d(x0,y0,>bar,grid=0);
    h=holding(1);
    frame();
    xgrid(ticks(x));
    p=plot();
    for i=-10 to 10;
      if i<=p[4] and i>=p[3] then
         ygrid(i,yt="10^"+i);
      endif;
    end;
    holding(h);
  endfunction
```

Mari kita uji dengan nilai yang terdistribusi secara eksponensial.

```
>aspect(1.5); x=1:10; y=-log(random(size(x)))*200; ...
>logimpulseplot1(x,y):
```



Mari kita menghidupkan kurva 2D dengan menggunakan plot langsung. Perintah plot(x,y) hanya memplot kurva ke dalam jendela plot. setplot(a,b,c,d) mengatur jendela ini.

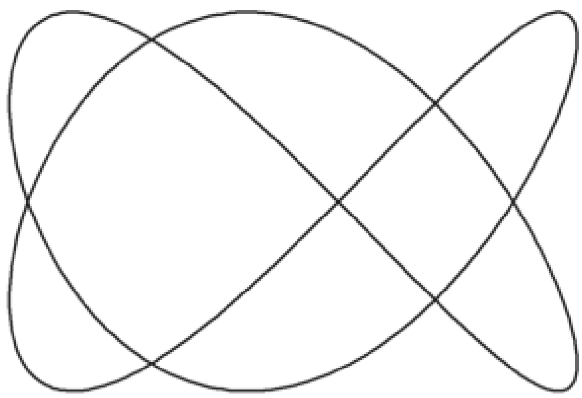
Fungsi wait(0) memaksa plot untuk muncul pada jendela grafis. Kalau tidak, penggambaran ulang dilakukan dalam interval waktu yang jarang.

```
>function animliss (n,m) ...
```

```
t=linspace(0,2pi,500);
f=0;
c=framecolor(0);
l=linewidth(2);
setplot(-1,1,-1,1);
repeat
  clg;
  plot(sin(n*t),cos(m*t+f));
  wait(0);
  if testkey() then break; endif;
  f=f+0.02;
end;
framecolor(c);
linewidth(l);
endfunction
```

Tekan sembarang tombol untuk menghentikan animasi ini.

```
>animliss(2,3): // lihat hasilnya, jika sudah puas, tekan ENTER
```

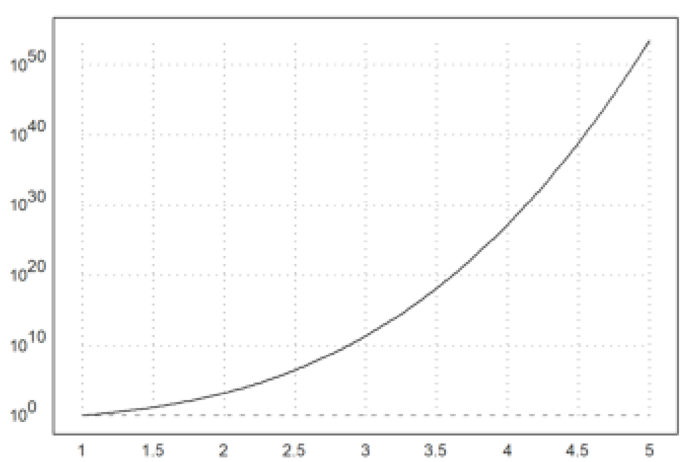


Plot Logaritmik

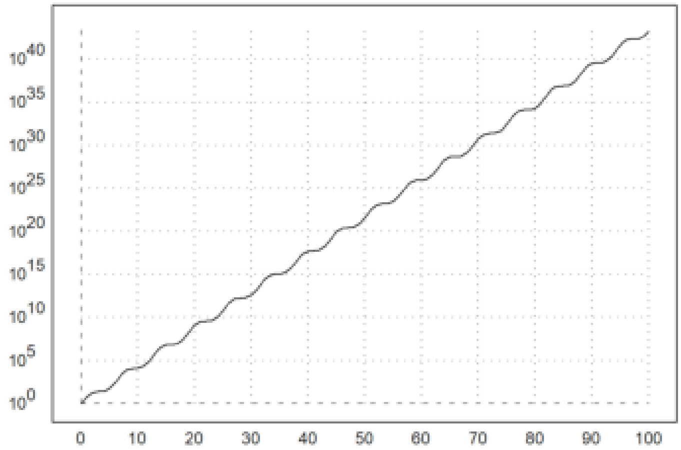
EMT menggunakan parameter "logplot" untuk skala logaritmik.
Plot logaritmik dapat diplot menggunakan skala logaritmik dalam y dengan logplot = 1, atau menggunakan skala logaritmik dalam x dan y dengan logplot = 2, atau dalam x dengan logplot = 3.

```
- logplot=1: y-logarithmic
- logplot=2: x-y-logarithmic
- logplot=3: x-logarithmic
```

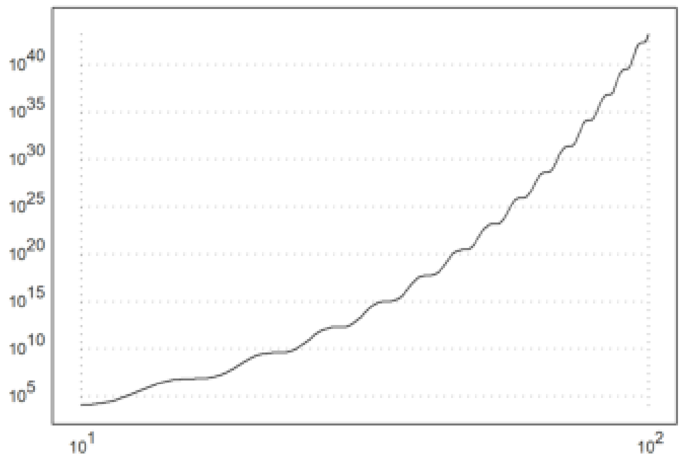
```
>plot2d("exp(x^3-x)*x^2",1,5,logplot=1):
```



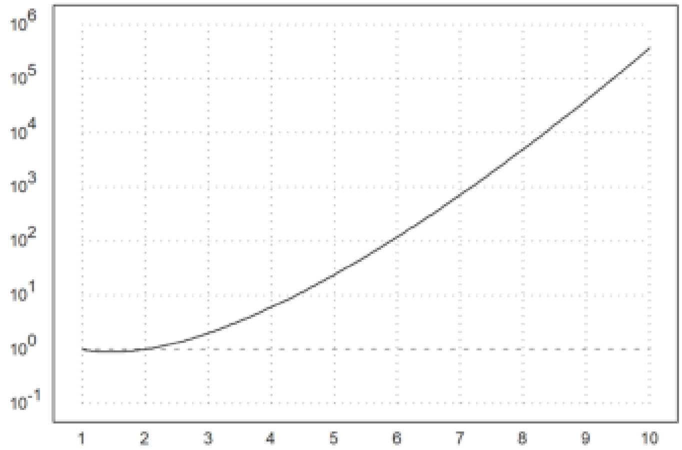
```
>plot2d("exp(x+sin(x))",0,100,logplot=1):
```



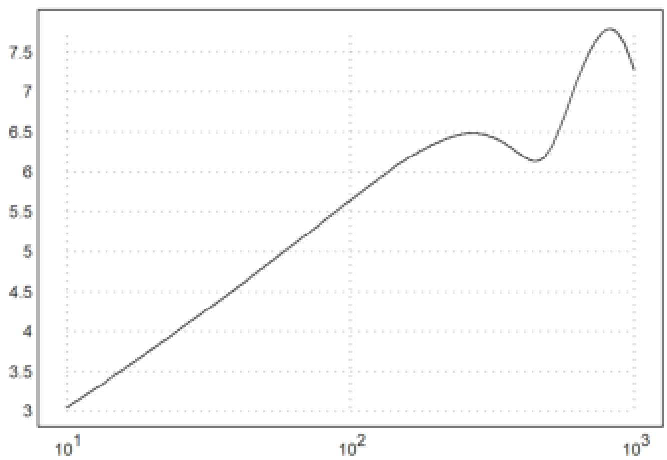

```
>plot2d("exp(x+sin(x))",10,100,logplot=2):
```




```
>plot2d("gamma(x)",1,10,logplot=1):
```

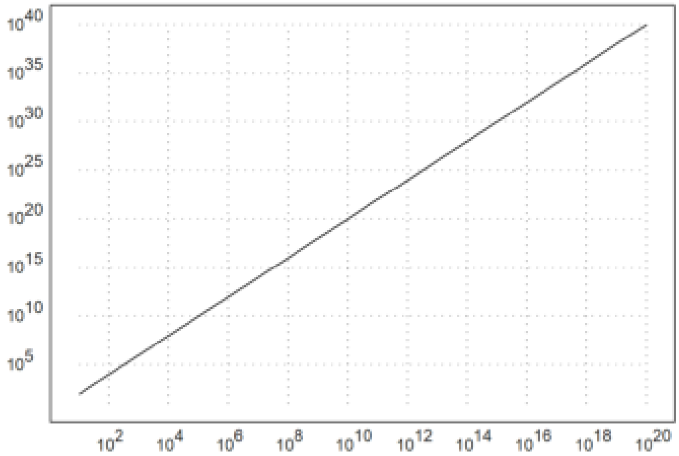



```
>plot2d("log(x*(2+sin(x/100)))",10,1000,logplot=3):
```



Hal ini juga berlaku pada plot data.

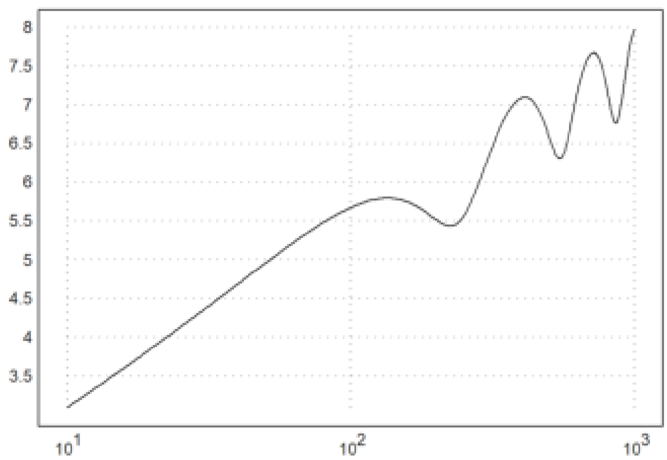
```
>x=10^(1:20); y=x^2-x;
>plot2d(x,y,logplot=2):
```



Contoh Tambahan:
Tentukan plot dari fungsi berikut!

$$log(x(2 + sin(2x/100))$$

```
>plot2d("log(x*(2+sin(2*x/100)))",10,1000,logplot=3):
```



Rujukan Lengkap Fungsi plot2d()

```
function plot2d (xv, yv, btest, a, b, c, d, xmin, xmax, r, n,  ..
logplot, grid, frame, framecolor, square, color, thickness, style, ..
auto, add, user, delta, points, addpoints, pointstyle, bar, histogram,  ..
distribution, even, steps, own, adaptive, hue, level, contour,  ..
nc, filled, fillcolor, outline, title, xl, yl, maps, contourcolor, ..
contourwidth, ticks, margin, clipping, cx, cy, insimg, spectral,  ..
cgrid, vertical, smaller, dl, niveau, levels)
```

Multipurpose plot function for plots in the plane (2D plots). This function can do plots of functions of one variables, data plots, curves in the plane, bar plots, grids of complex numbers, and implicit plots of functions of two variables.

Parameters
x,y : equations, functions or data vectors
a,b,c,d : Plot area (default a=-2,b=2)
r : if r is set, then a=cx-r, b=cx+r, c=cy-r, d=cy+r

```
r can be a vector [rx,ry] or a vector [rx1,rx2,ry1,ry2].
```

xmin,xmax : range of the parameter for curves
auto : Determine y-range automatically (default)
square : if true, try to keep square x-y-ranges
n : number of intervals (default is adaptive)
grid : 0 = no grid and labels,

```
1 = axis only,
2 = normal grid (see below for the number of grid lines)
3 = inside axis
4 = no grid
5 = full grid including margin
6 = ticks at the frame
7 = axis only
8 = axis only, sub-ticks
```

frame : 0 = no frame
framecolor: color of the frame and the grid
margin : number between 0 and 0.4 for the margin around the plot
color : Color of curves. If this is a vector of colors,

```
it will be used for each row of a matrix of plots. In the case
point plots, it should be a column vector. If a row vector or a
full matrix of colors is used for point plots, it will be used
each data point.
```

thickness : line thickness for curves

```
This value can be smaller than 1 for very thin lines.
```

style : Plot style for lines, markers, and fills.

```
For points use
"[]", "<>", ".", "..", "...",
"*", "+", "|", "-", "o"
"[]#", "<>#", "o#" (filled shapes)
"[]w", "<>w", "ow" (non-transparent)
For lines use
"-", "--", "-.", ".", ".-.", "-.-", "->"
For filled polygons or bar plots use
"#", "#O", "O", "/", "\", "\/",
"+", "|", "-", "t"
```

points : plot single points instead of line segments
addpoints : if true, plots line segments and points
add : add the plot to the existing plot
user : enable user interaction for functions
delta : step size for user interaction
bar : bar plot (x are the interval bounds, y the interval values)
histogram : plots the frequencies of x in n subintervals
distribution=n : plots the distribution of x with n subintervals
even : use inter values for automatic histograms.
steps : plots the function as a step function (steps=1,2)
adaptive : use adaptive plots (n is the minimal number of steps)
level : plot level lines of an implicit function of two variables
outline : draws boundary of level ranges.
If the level value is a 2xn matrix, ranges of levels will be drawn
in the color using the given fill style. If outline is true, it
will be drawn in the contour color. Using this feature, regions of
f(x,y) between limits can be marked.
hue : add hue color to the level plot to indicate the function

```
value
```

contour : Use level plot with automatic levels
nc : number of automatic level lines
title : plot title (default "")
xl, yl : labels for the x- and y-axis
smaller : if >0, there will be more space to the left for labels.
vertical :

```
Turns vertical labels on or off. This changes the global variable
verticallabels locally for one plot. The value 1 sets only vertical
text, the value 2 uses vertical numerical labels on the y axis.
```

filled : fill the plot of a curve
fillcolor : fill color for bar and filled curves
outline : boundary for filled polygons
logplot : set logarithmic plots

```
1 = logplot in y,
2 = logplot in xy,
3 = logplot in x
```

own :

```
A string, which points to an own plot routine. With >user, you get
the same user interaction as in plot2d. The range will be set
before each call to your function.
```

maps : map expressions (0 is faster), functions are always mapped.
contourcolor : color of contour lines
contourwidth : width of contour lines
clipping : toggles the clipping (default is true)
title :

```
This can be used to describe the plot. The title will appear above
the plot. Moreover, a label for the x and y axis can be added with
xl="string" or yl="string". Other labels can be added with the
functions label() or labelbox(). The title can be a unicode
string or an image of a Latex formula.
```

cgrid :

```
Determines the number of grid lines for plots of complex grids.
Should be a divisor of the the matrix size minus 1 (number of
subintervals). cgrid can be a vector [cx,cy].
```

Overview

The function can plot

- expressions, call collections or functions of one variable,
- parametric curves,
- x data against y data,
- implicit functions,
- bar plots,
- complex grids,
- polygons.

If a function or expression for xv is given, plot2d() will compute values in the given range using the function or expression. The expression must be an expression in the variable x. The range must be defined in the parameters a and b unless the default range should be used. The y-range will be computed automatically, unless c and d are specified, or a radius r, which yields the range
r,r
for x and y. For plots of functions, plot2d will use an adaptive evaluation of the function by default. To speed up the plot for complicated functions, switch this off with <adaptive, and optionally decrease the number of intervals n. Moreover, plot2d() will by default use mapping. I.e., it will compute the plot element for element. If your expression or your functions can handle a vector x, you can switch that off with <maps for faster evaluation.

Note that adaptive plots are always computed element for element. If functions or expressions for both xv and for yv are specified, plot2d() will compute a curve with the xv values as x-coordinates and the yv values as y-coordinates. In this case, a range should be defined for the parameter using xmin, xmax. Expressions contained in strings must always be expressions in the parameter variable x.

BAB 3

PRESENTASI SUBTOPIK 1 DAN 2 MATERI PLOT 3D DENGAN EMT

Nama : Wahyu Rananda Westri
NIM : 22305144039
Kelas : Matematika B
Kelompok : 3

URAIAN MATERI

DEFINISI FUNGSI

Sebuah fungsi f adalah suatu aturan padanan yang menghubungkan tiap obyek x dalam satu himpunan, yang disebut daerah asal, dengan sebuah nilai unik f(x) dari himpunan kedua. Himpunan nilai yang diperoleh secara demikian disebut daerah hasil.

DEFINISI FUNGSI DUA VARIABEL

Sebuah fungsi bernilai-riil dari dua variabel riil; yakni, fungsi f yang memadankan setiap pasangan terurut (x,y) pada suatu himpunan D dari bidang dengan satu dan hanya satu bilangan real yang ditulis sebagai z = f (x,y).

Himpunan D disebut daerah asal fungsi. Sedangkan daerah nilai fungsi adalah himpunan nilai-nilainya. Misalnya z = f (x,y), merupakan fungsi dua variabel dengan x,y disebut sebagai variabel bebas(independent variable) dan z variabel tak bebas (dependent variable). Sebagai contoh

$$z = f(x, y) = x^2 + 2y^3$$

Perhatikan grafik fungsi permukaan bola dengan persamaan

$$x^2 + y^2 + z^2 = 1$$

yang berpusat di titik asal O(0,0,0) dan berjari-jari 1. Dalam permukaan tersebut titik-titik (x,y)=(0,0) berpadanan dengan dua nilai z, yakni -1 dan 1. Artinya oleh permukaan tersebut terdapat pemetaan dari(0,0) ke dua nilai berbeda, maka pemetaan seperti itu bukan merupakan suatu fungsi.

PERMUKAAN DALAM R^3(RUANG DIMENSI 3)

Terdapat tiga sumbu koordinat yang saling tegak lurus yaitu: sumbu x, sumbu y, sumbu z. Ruang R^3 oleh ketiga sumbu x,y,z tersekat dalam delapan oktan. Kumpulan dari titik-titik di R^3 dapat berupa kurva ataupun permukaan. Dalam R^3 terdapat permukaan linear(berupa bidang datar) dan kuadratik(berupa bidang lengkung). Permukaan linear tidak mungkin dapat dibuat keseluruhan bidangnya, cukup digambar wakil bidang yang dapat berupa segitiga, segiempat, dll. Permukaan kuadratik dapat berupa permukaan bola, elipsoida, paraboloida, tabung elips, tabung lingkaran, atau tabung parabola. Beberapa persamaan umum dari permukaan kuadratik.
- Bola:

$$x^2 + y^2 + z^2 = a^2, a > 0$$

- Elipsoida:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1, a, b, c > 0$$

- Hiperboloida Berdaun Satu:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 1, a, b, c > 0$$

- Hiperboloida Berdaun Dua:

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} - \frac{z^2}{c^2} = 1, a, b, c > 0$$

- Paraboloida Eliptik:

$$z = \frac{x^2}{a^2} + \frac{y^2}{b^2}, a, b > 0$$

- Paraboloida Hiperbolik:

$$z = \frac{y^2}{b^2} - \frac{x^2}{a^2}, a, b > 0$$

- Kerucut Eliptik:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} - \frac{z^2}{c^2} = 0$$

Dalam menggambar sketsa permukaan, dapat dibuat langkah bantuan dengan menggambar perpotongan permukaan tersebut dengan tiga bidang utama, yaitu XOY, XOZ, dan YOZ. Jika permukaan sangat rumit,dapat digunakan komputer untuk menggambar grafiknya,

GRAFIK FUNGSI DUA VARIABEL

Grafik fungsi dua variabel adalah himpunan

$$(x, y, z)|z = f(x, y), (x, y) \in D$$

yang merupakan himpunan titik dalam ruang atau R3. Himpunan ini pada umumnya membentuk permukaan di ruang. Ketika kita menyebut grafik dari fungsi f dengan dua variabel, yang dimaksud adalah grafik dari persamaan z = f (x,y).

Beberapa fungsi matematika yang terlibat dalam menggambar grafik fungsi dua varibel.
1. Fungsi Linear
Bentuk umum

$$f(x, y) = ax + by + c$$

di mana a, b, dan c adalah konstanta. Grafiknya adalah bidang datar.
Contoh:

$$f(x, y) = 2x + 5y + 3$$

2. Fungsi Kuadratik
Bentuk umum

$$f(x, y) = ax^2 + by^2 + cxy + dx + ey + f.$$

dimana a, b, c, d, e, dan f adalah konstanta. Grafik fungsi kuadrat ini adalah sebuah permukaan yang dapat memiliki berbagai bentuk tergantung pada nilai-nilai konstantanya.
Contoh:

$$f(x, y) = 2x^2 - 4y^2 + 3xy$$

3.Fungsi Trigonometri
Fungsi trigonometri dua variabel adalah fungsi matematika yang melibatkan operasi trigonometri (seperti sin, cos, tan) pada kedua variabel x dan y. Contoh:

$$f(x, y) = \sin x . \cos y$$

4.Fungsi Aljabar
Fungsi aljabar adalah fungsi yang bisa didefinisikan sebagai akar dari sebuah persamaan aljabar. Fungsi aljabar merupakan ekspresi aljabar menggunakan sejumlah suku terbatas, yang melibatkan operasi aljabar seperti penambahan, pengurangan, perkalian, pembagian, dan peningkatan menjadi pangkat pecahan. Contoh dari fungsi tersebut adalah:

$$f(x, y) = 1/xy$$

$$f(x, y) = \sqrt{xy}$$

$$f(x, y) = \frac{\sqrt{1 + x^3}}{3^{3/7} - \sqrt{7}y^{1/3}}$$

5.Fungsi Eksponensial
Fungsi eksponensial dua variabel bisa dinyatakan

$$f(x, y) = a.b^{xy}$$

dimana a dan b adalah konstanta, x dan y adalah variabel. Fungsi ini menggambarkan pertumbuhan eksponensial yang bergantung pada nilai x dan y.
Contoh:

$$f(x, y) = 2.3^{xy}$$

1. Menggambar Grafik Fungsi Dua Variabel dalam Bentuk Ekspresi

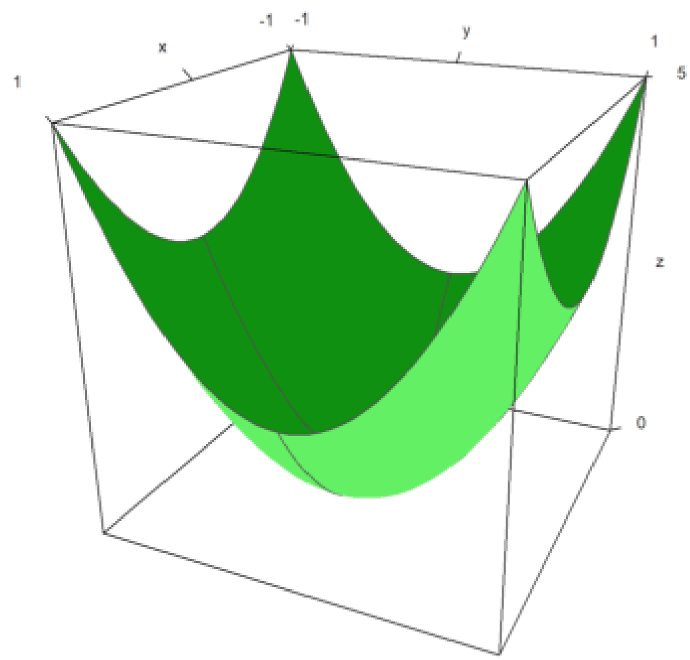
Langsung
Grafik fungsi dua variabel dalam bentuk ekspresi langsung adalah representasi visual dari hubungan matematis antara dua variabel independen yang dinyatakan dalam bentuk persamaan atau ekspresi matematis.

Contoh Soal 1(Fungsi Kuadratik)

Gambarlah grafik dari fungsi berikut.

$$f(x,y) = 2x^2 + 3y^2$$

```
>plot3d("2*x^2+3*y^2",n=40,grid=2):
```



Gambar di atas menampilkan grafik fungsi dengan n=40 dan grid=2.
- n = jumlah interval kisi-kisi,jumlah n default=40
- grid = jumlah garis kisi di setiap arah, jumlah grid default=10

Penjelasan:
misalkan

$$z = 2x^2 + 3y^2$$

$$z = \frac{x^2}{\frac{1}{2}} + \frac{y^2}{\frac{1}{3}}$$

(yang dikenal sebagai persamaan sebuah paraboloida eliptik)
dan perhatikan bahwa

$$z \geq 0$$

cari jejak pada bidang koordinat
-bidang XOY(z=0):

$$\frac{x^2}{\frac{1}{2}} + \frac{y^2}{\frac{1}{3}} = 0$$

jika z=0 maka x^2 dan y^2 juga harus 0, maka diperoleh titik (0,0,0)

-bidang YOZ(x=0)

$$\frac{y^2}{\frac{1}{3}} = z$$

$$y^2 = \frac{1}{3}z$$

(berupa parabola searah sumbu z dan titik puncaknya (0,0))

-bidang XOZ(y=0)

$$\frac{x^2}{\frac{1}{2}} = z$$

$$x^2 = \frac{1}{2}z$$

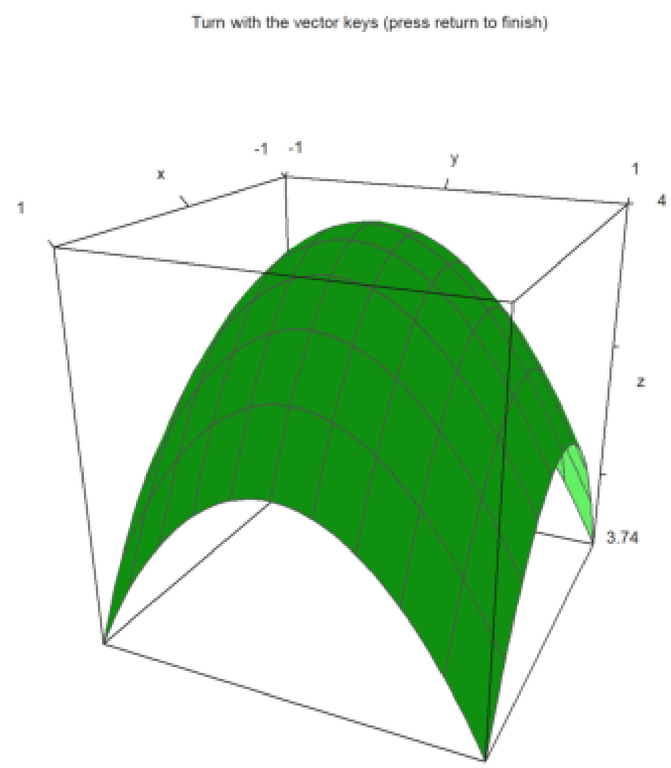
(berupa parabola searah sumbu z dan titik puncaknya (0,0))

Contoh Soal 2(Fungsi Aljabar)

Gambarlah grafik dari fungsi berikut.

$$f(x,y) = \sqrt{16 - (x^2 + y^2)}$$

```
>plot3d("(16-x^2-y^2)^(1/2)",>user, ...
>title= "Turn with the vector keys (press return to finish)"):
```



Gambar di atas menampilkan grafik fungsi dengan menggunakan >user.
Untuk menggunakan >user, kita dapat menekan tombol:
- kiri,kanan,atas,bawah:putar sudut pandang
- +-:memperbesar atau memperkecil
- a:menghasilkan anaglyph
- l:sakelar untuk memutar sumbu cahaya
- spasi:setel ulang ke default
- enter: mengakhiri interaksi

Penjelasan :
misalkan

$$z = \sqrt{16 - (x^2 + y^2)}$$

dan perhatikan bahwa

$$z \geq 0$$

Jika kedua ruas dikuadratkan dan sederhanakan, akan kita peroleh persamaan

$$x^2 + y^2 + z^2 = 16$$

yang kita kenal sebagai persamaan sebuah bola.

cari jejak pada bidang koordinat
-bidang XOY(z=0):

$$x^2 + y^2 = 16$$

(berupa lingkaran dengan pusat(0,0) dan jari-jari 4)
-bidang YOZ(x=0)

$$y^2 + z^2 = 16$$

(berupa lingkaran dengan pusat(0,0) dan jari-jari 4)
-bidang XOZ(y=0)

$$x^2 + z^2 = 16$$

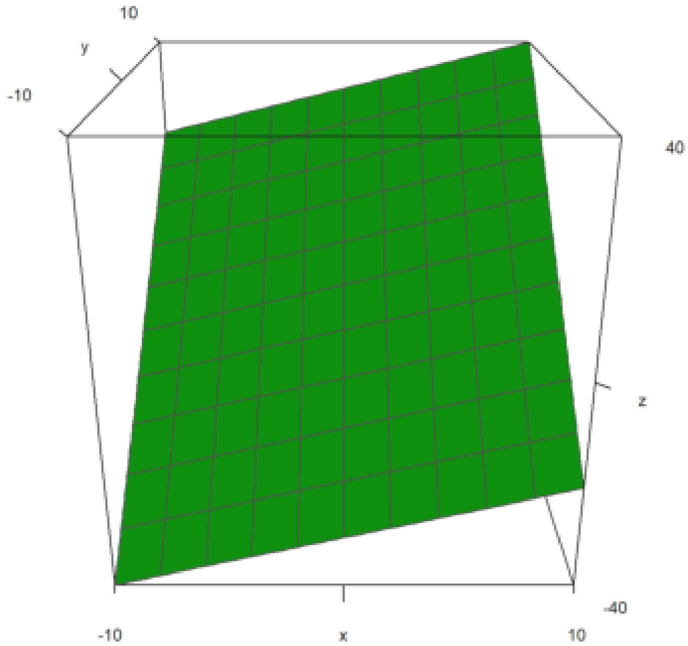
(berupa lingkaran dengan pusat(0,0) dan jari-jari 4)

Contoh Soal 3 (Fungsi Linear)

Gambarlah grafik dari fungsi berikut.

$$f(x, y) = x + 3y$$

```
>  plot3d("x+3*y",angle=0°,a=-10,b=10,c=-10,d=10,fscale=10):
```



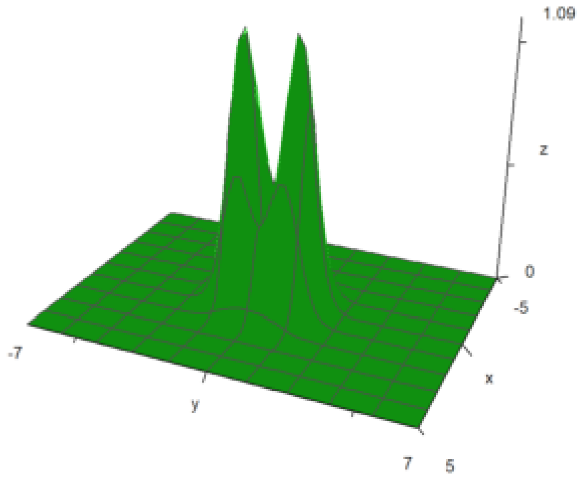
Gambar di atas menampilkan grafik fungsi dengan angle=0 derajat,a=-10,b=10,c=-10,d=10,fscale=10.
- angle: sudut pandang
- a,b: rentang x
- c,d: rentang y
- fscale: skala ke nilai fungsi (defaultnya adalah <fscale)

Contoh Soal 4(Fungsi Eksponensial)

Gambarlah grafik dari fungsi berikut.

$$f(x,y) = (x^2 + 3y^2)e^{-x^2-y^2}$$

```
>plot3d("(x^2+3*y^2)*E^(-x^2-y^2)",scale={1,2},xmin=-5,xmax=5,ymin=-7,ymax=
```



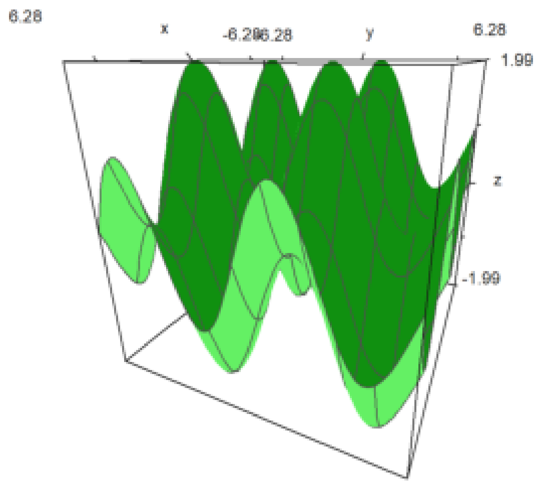
Gambar di atas menampilkan grafik fungsi dengan scale=[1,2],xmin=-5,xmax=5,ymin=-7,ymax=7,fram
- scale: angka atau vektor 1x2 untuk menskalakan ke arah x dan y
- xmin,xmax: rentang x
- ymin,ymax: rentang y
- frame: jenis bingkai (default 1)

Contoh Soal 5(Fungsi Trigonometri)

Gambarlah grafik dari fungsi berikut.

$$f(x,y) = \sin x + \sin y$$

```
>plot3d("sin(x)+sin(y)",r=2*pi,distance=3,zoom=1,center=[0.1,0,0],height=20
```



Gambar di atas menampilkan grafik fungsi dengan r=2pi,distance=3,zoom=1,center=[0.1,0,0],height=2
derajat.
- r: dapat digunakan sebagai ganti xmin, xmax, ymin, ymax; r dapat berupa vektor [rx, ry] atau [rx, ry, rz]
- distance: jarak pandang plot
- zoom: nilai zoom
- center: memindahkan bagian tengah plot
- height: ketinggian tampilan dalam radian

Nilai default dari distance, zoom, angle, height dapat diperiksa atau diubah dengan fungsi view. Fungsi ini mengembalikan parameter sesuai urutan di atas.

```
>view
```

```
[5,  2.6,  2,  0.4]
```

2. Menggambar Grafik Fungsi Dua Variabel yang Rumusnya Disimpan

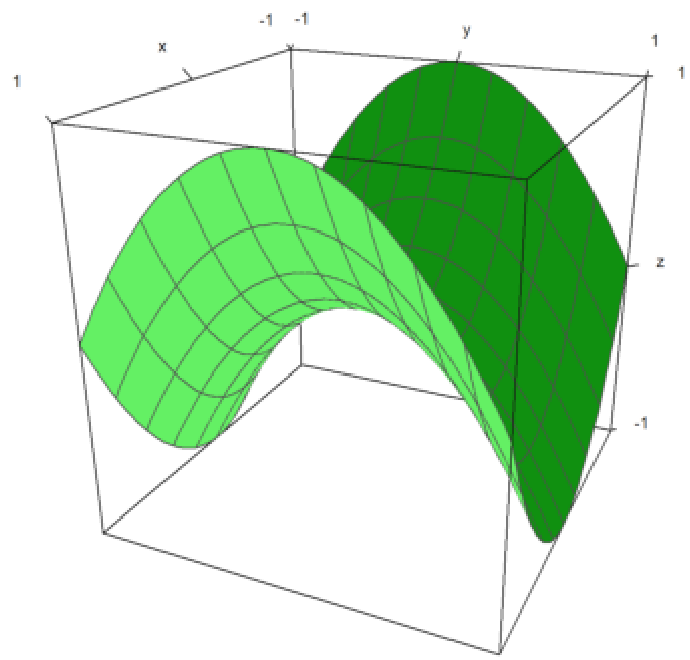
dalam Variabel Ekspresi

Untuk menyimpan sebuah fungsi, dapat dilakukan menggunakan perintah function. Kemudian untuk memanggil atau membuat grafiknya tinggal memanggil nama fungsi tersebut. Contohnya :

```
>function b(x,y):=x^2-y^2;
```

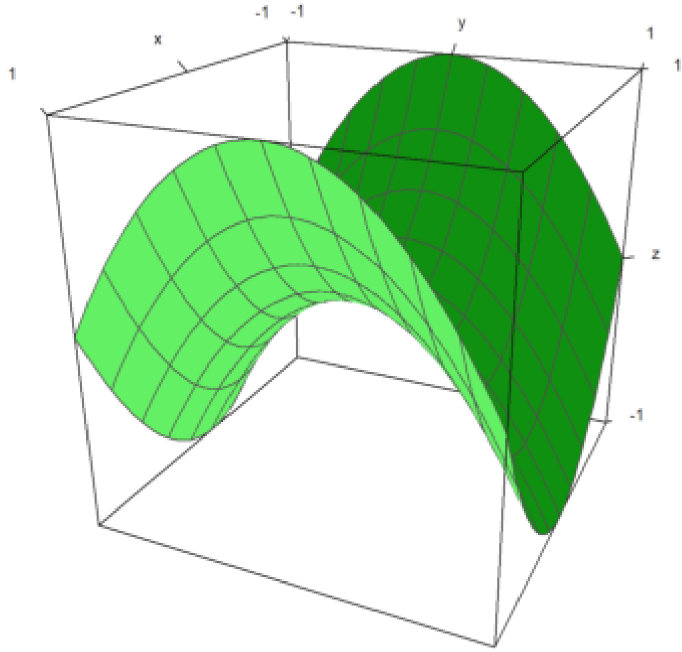
selanjutnya kita akan membuat grafik dari fungsi tersebut

```
>plot3d("b(x,y)"):
```



selain cara di atas, kita juga bisa membuat grafik dari fungsi tersebut dengan

```
>plot3d("b"):
```

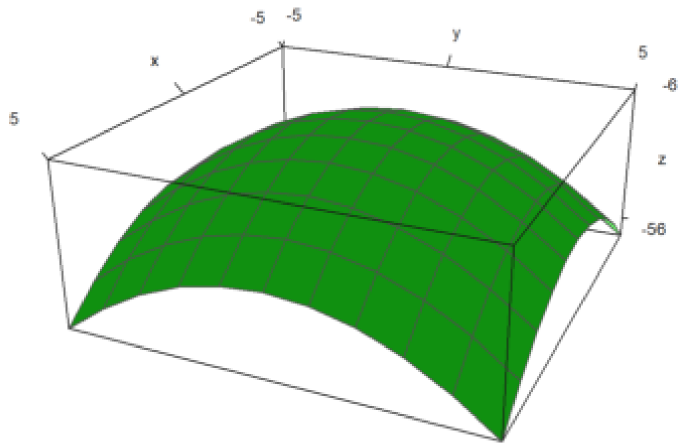


Contoh Soal 1(Fungsi Kuadratik)

Gambarlah grafik dari fungsi tersebut.

$$f(x,y) = -6 - x^2 - y^2$$

```
>function p(x,y):=-6-x^2-y^2;
>plot3d("p(x,y)",r=5, ...
>fscale=2,n=10,zoom=2.7):
```



Penjelasan :
misalkan

$$z = -6 - x^2 - y^2$$

Cari domainnya

$$D_z = [(x, y)|x, y \in R^2]$$

Cari daerah hasilnya

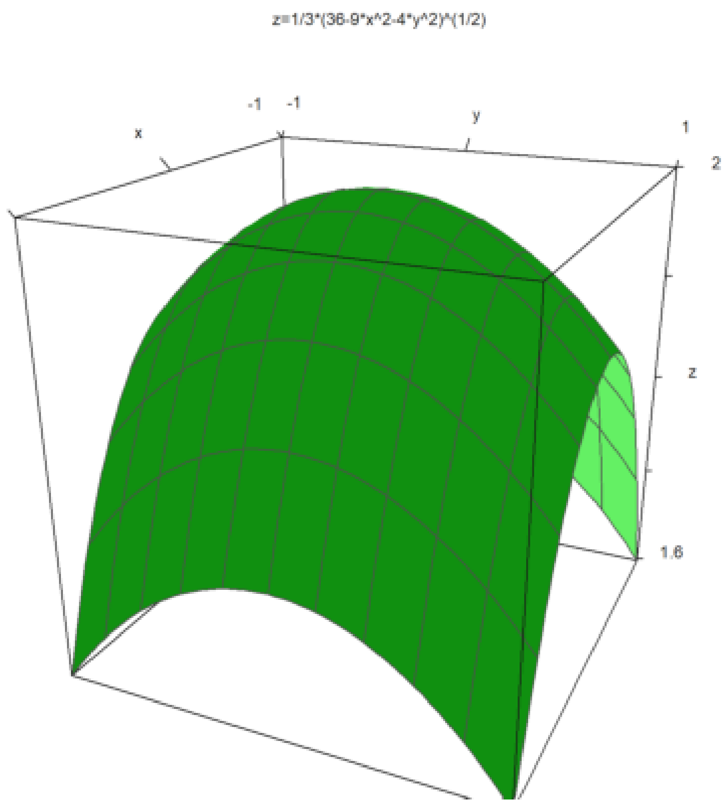
$$R_z = [-\infty, -6]$$

Contoh Soal 2(Fungsi Aljabar)

Gambarlah grafik dari fungsi berikut.

$$f(x, y) = \frac{1}{3}\sqrt{36 - 9x^2 - 4y^2}$$

```
>function z(x,y) :=1/3*(36-9*x^2-4*y^2)^(1/2);
>plot3d("z(x,y)",title="z=1/3*(36-9*x^2-4*y^2)^(1/2)",zoom=3):
```



Penjelasan :
misalkan

$$z = \frac{1}{3}\sqrt{36 - 9x^2 - 4y^2}$$

dan perhatikan bahwa

$$z \geq 0$$

Jika kedua ruas di kuadratkan dan sederhanakan, akan diperoleh:

$$9x^2 + 4y^2 + 9z^2 = 36$$

yang dikenal sebagai persamaan sebuah elipsoida.
cari jejak pada bidang koordinat
-bidang XOY(z=0):

$$9x^2 + 4y^2 = 36$$

$$\frac{x^2}{4} + \frac{y^2}{9} = 1$$

(berupa elips dengan pusat(0,0), titik puncak : (0,-2),(0,2),(0,3),(0,-3))

-bidang YOZ(x=0)

$$4y^2 + 9z^2 = 36$$

$$\frac{y^2}{9} + \frac{z^2}{4} = 1$$

(berupa elips dengan pusat (0,0), titik puncak : (0,-3),(0,3),(0,-2),(0,2))

-bidang XOZ(y=0)

$$9x^2 + 9z^2 = 36$$

$$x^2 + z^2 = 4$$

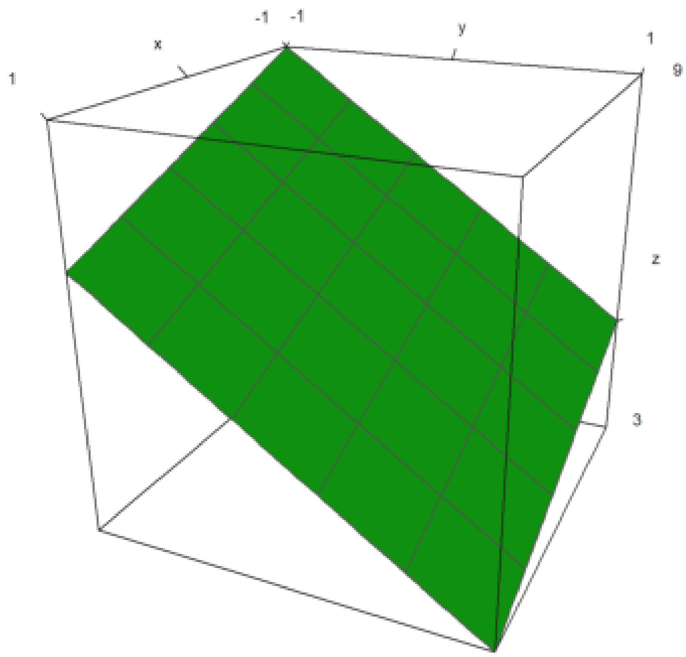
(berupa lingkaran dengan pusat(0,0), jari-jari=2)

Contoh Soal 3(Fungsi Linear)

a) Gambarlah grafik dari fungsi tersebut.

$$f(x, y) = 6 - x - 2y$$

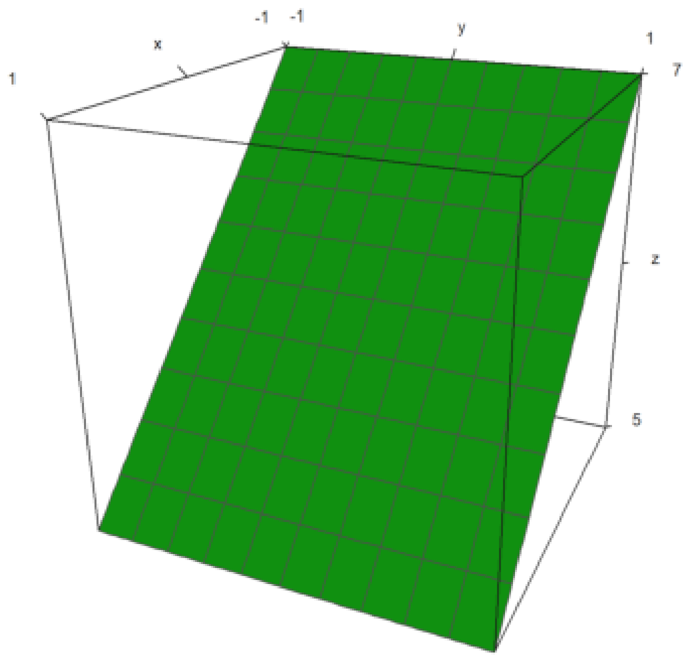
```
>function q(x,y):=6-x-2*y;
>plot3d("q(x,y)",grid=5):
```



b) Gambarlah grafik dari fungsi tersebut.

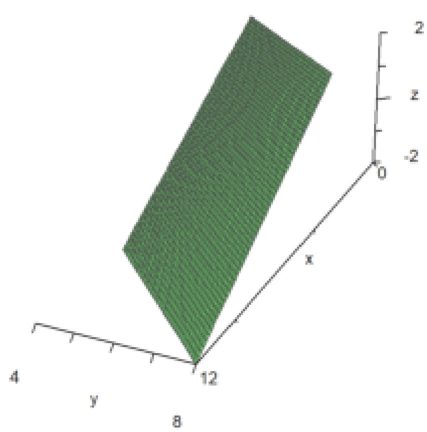
$$f(x, y) = 6 - x$$

```
>function j(x,y):=6-x;
>plot3d("j"):
```



Berikut adalah plot dari tiga fungsi.

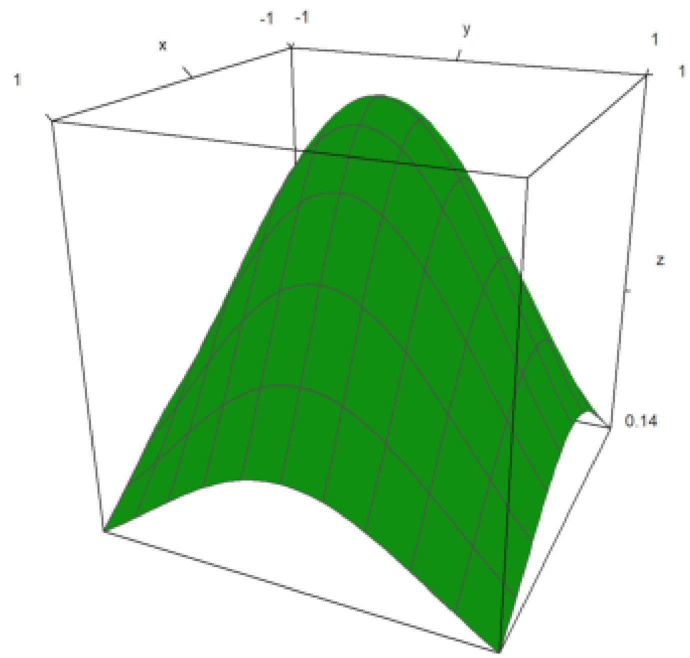
```
>plot3d("q","j","y",r=2,zoom=3,frame=3):
```



a) Gambarlah grafik dari fungsi tersebut.

$$f(x, y) = e^{-(x^2+y^2)}$$

```
>function n(x,y):= E^(-(x^2+y^2));
>plot3d("n",>fscale,>scale):
```



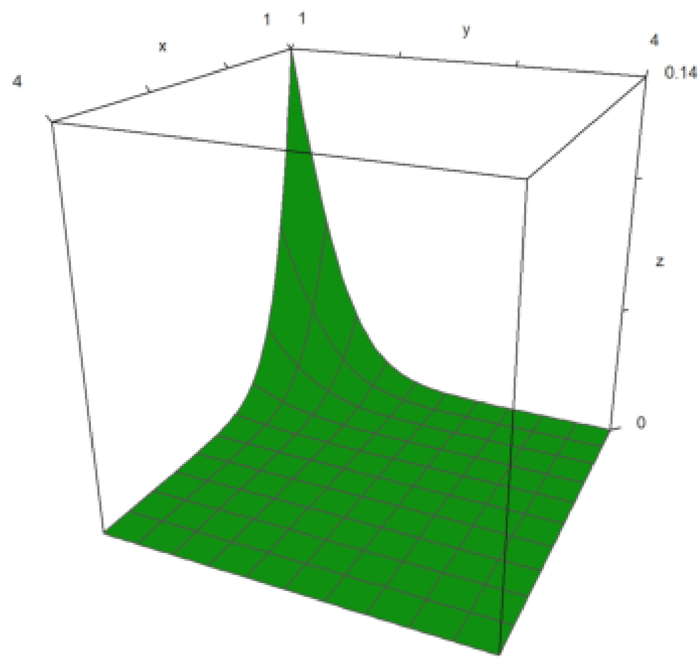
b) Gambarlah grafik dari fungsi pada contoh soal a dengan syarat:

$$1 \leq x \leq 4$$

dan

$$1 \leq y \leq 4$$

```
>plot3d("n",a=1,b=4,c=1,d=4):
```

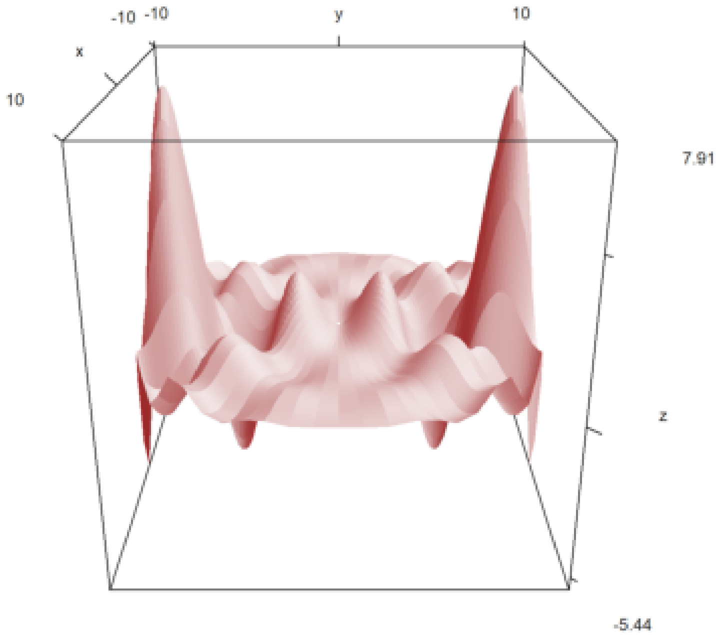


Contoh Soal 5(Fungsi Trigonometri)

Gambarlah grafik dari fungsi tersebut.

$$f(x,y) = \frac{\sin x \sin y}{xy}$$

```
>function m(x,y):=(sin(x)*sin(y))/x*y;
>plot3d("m",r=10,angle=90°,fscale=-1,>user,>polar,color=red,>hue):
```



Dalam membuat grafik di atas terdapat >polar.
- >polar: menghasilkan plot polar
- hue: mengaktifkan bayangan cahaya
- color: mengatur warna pada grafik

BAB 4

MENGGAMBAR PLOT 3D DENGAN EMT

Menggambar Plot 3D dengan EMT

Ini adalah pengenalan plot 3D di Euler. Kita memerlukan plot 3D untuk memvisualisasikan fungsi dari dua variabel.

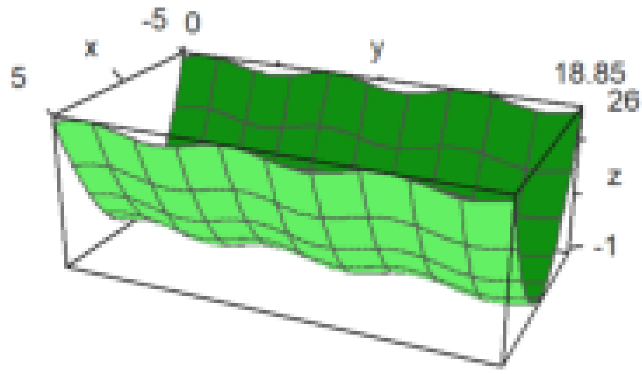
Euler menggambar fungsi-fungsi tersebut dengan menggunakan algoritme pengurutan untuk menyembunyikan bagian-bagian di latar belakang. Secara umum, Euler menggunakan proyeksi pusat. Standarnya adalah dari kuadran x-y positif ke arah asal x=y=z=0, tetapi sudut=0° terlihat dari arah sumbu-y. Sudut pandang dan ketinggian dapat diubah.

Euler dapat memplot :

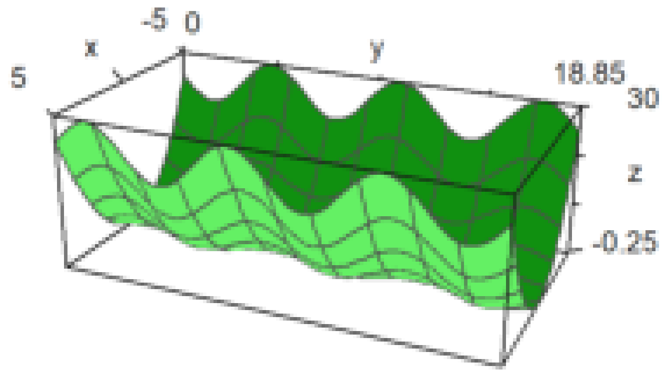
- memplot - permukaan dengan garis bayangan dan garis datar,
- awan titik,
- kurva parametrik,
- permukaan implisit.

Plot 3D suatu fungsi menggunakan plot3d. Cara termudah adalah dengan memplot ekspresi dalam x dan y. Parameter r mengatur rentang plot sekitar (0,0).

```
>aspect(1.5); plot3d("x^2+sin(y)",-5,5,0,6*pi):
```

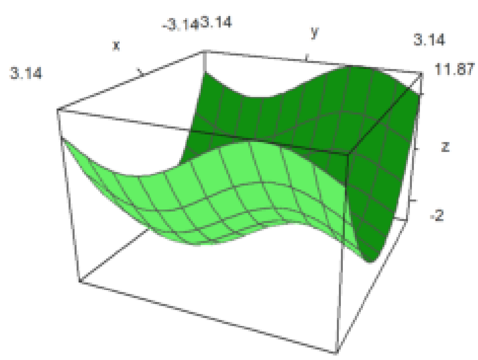



```
>plot3d("x^2+x*sin(y)",-5,5,0,6*pi):
```



Silakan lakukan modifikasi agar gambar "talang bergelombang" tersebut tidak lurus melainkan melengkung/melingkar, baik melingkar secara mendatar maupun melingkar turun/naik (seperti papan peluncur pada kolam renang. Temukan rumusnya.

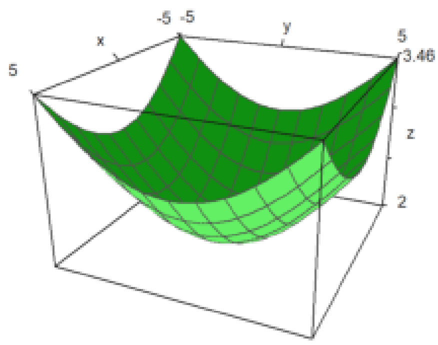
```
>aspect(1.5); plot3d("x^2+(2*sin(y))",r=pi):
```



Contoh soal
1. Gambarlah grafik dari fungsi tersebut.

$$f(x,y)=\sqrt{\frac{4x^2+4y^2+100}{25}}$$

```
>plot3d("((4*x^2+4*y^2+100)/25)^(1/2)",-5,5,-5,5):
```



Fungsi dua Variabel

Untuk grafik suatu fungsi, gunakan -

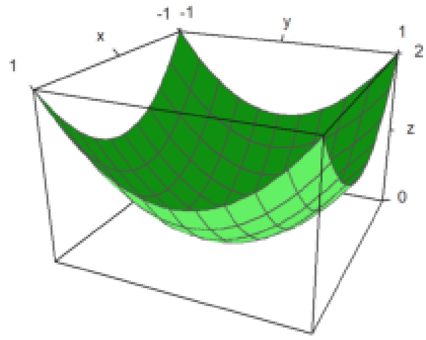
- ekspresi sederhana dalam x dan y,
- nama fungsi dari dua variabel,
- atau matriks data.

Standarnya adalah kisi-kisi kawat berisi dengan warna berbeda di kedua sisi. Perhatikan bahwa jumlah interval kisi default adalah 10, tetapi plot menggunakan jumlah default persegi panjang 40x40 untuk membuat permukaannya. Ini bisa diubah.

- n=40, n=[40,40]: jumlah garis grid di setiap arah.
- grid=10, grid=[10,10]: : jumlah garis grid di setiap arah.

Kami menggunakan default n=40 dan grid=10.

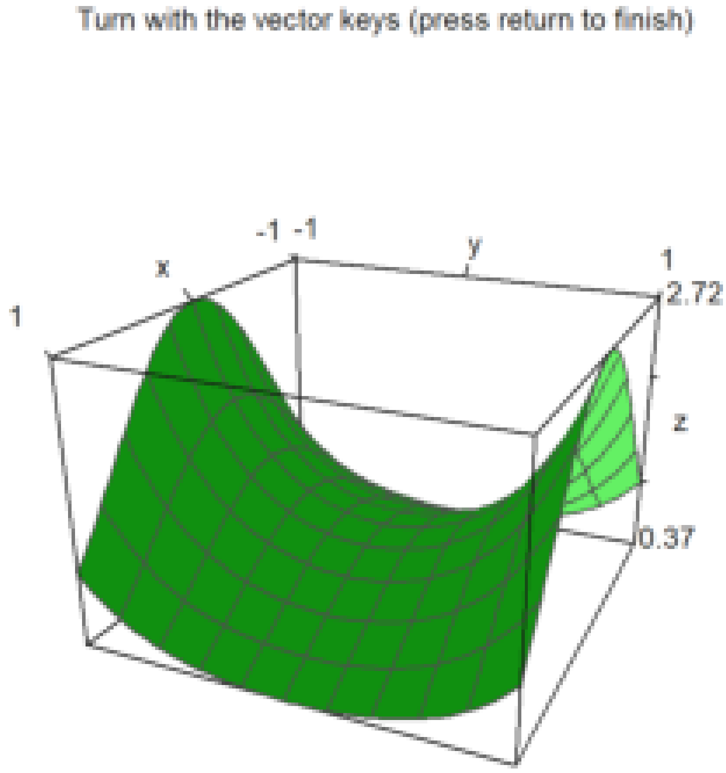
```
>plot3d("x^2+y^2"):
```



Interaksi pengguna dimungkinkan dengan parameter >pengguna. Pengguna dapat menekan tombol berikut.

- left,right,up,down: : putar sudut pandang,
- +,-: zoom in or out
- a: menghasilkan anaglyph (lihat di bawah)
- l: beralih memutar sumber cahaya(lihat di bawah)
- space: reset to default
- return: end interaction

```
>plot3d("exp(-x^2+y^2)",>user, ...
>  title="Turn with the vector keys (press return to finish)"):
```



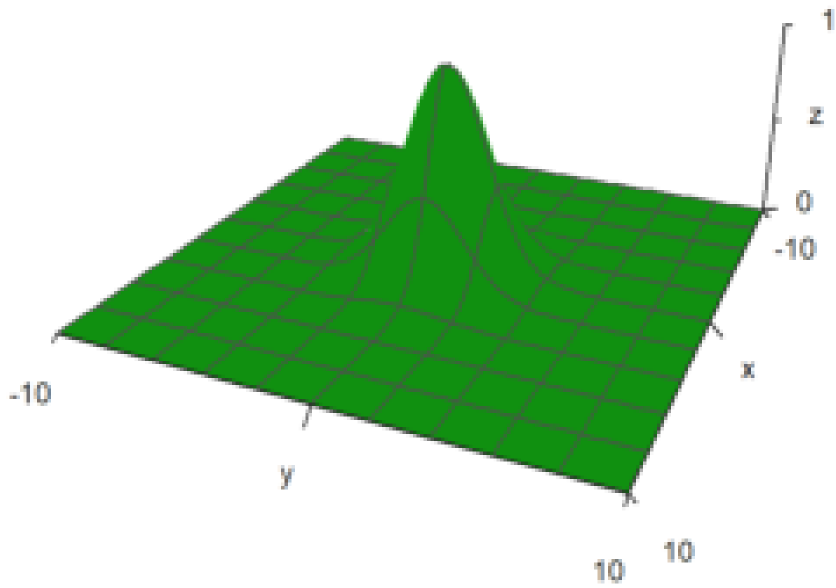
Rentang plot untuk fungsi dapat ditentukan dengan :

- a, b: rentang x
- c, d: rentang y
- r: bujur sangkar simetris di sekitar (0,0).
- n: jumlah subinterval untuk plot.

Ada beberapa parameter untuk menskalakan fungsi atau mengubah tampilan grafik.

fscale: skala untuk nilai fungsi (defaultnya adalah <fscale).
skala: Angka atau vektor 1x2 untuk menskalakan dalam arah x dan y
frame: jenis bingkai (standarnya adalah 1).

```
>plot3d("exp(-(x^2+y^2)/5)",r=10,n=80,fscale=4,scale=1.2,frame=3,>user):
```



Tampilan dapat diubah dengan berbagai cara.

- distance: jarak pandang ke plot.
- zoom: the zoom value.
- sudut: the angle to the negative y-axis in radians.
- height: the height of the view in radians.

Nilai default dapat diperiksa atau diubah dengan fungsi view(). Ini mengembalikan parameter dalam urutan di atas.

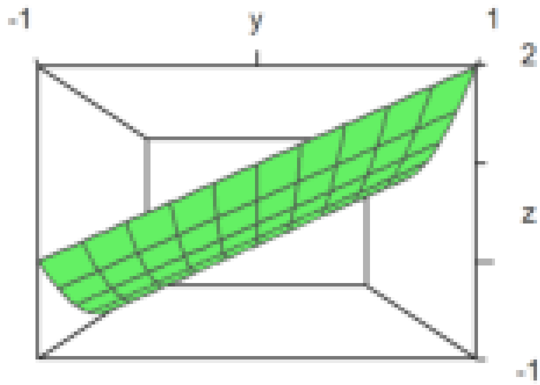
```
>view
```

```
[5,  2.6,  2,  0.4]
```

Jarak yang lebih dekat membutuhkan lebih sedikit zoom. Efeknya lebih seperti lensa sudut lebar.

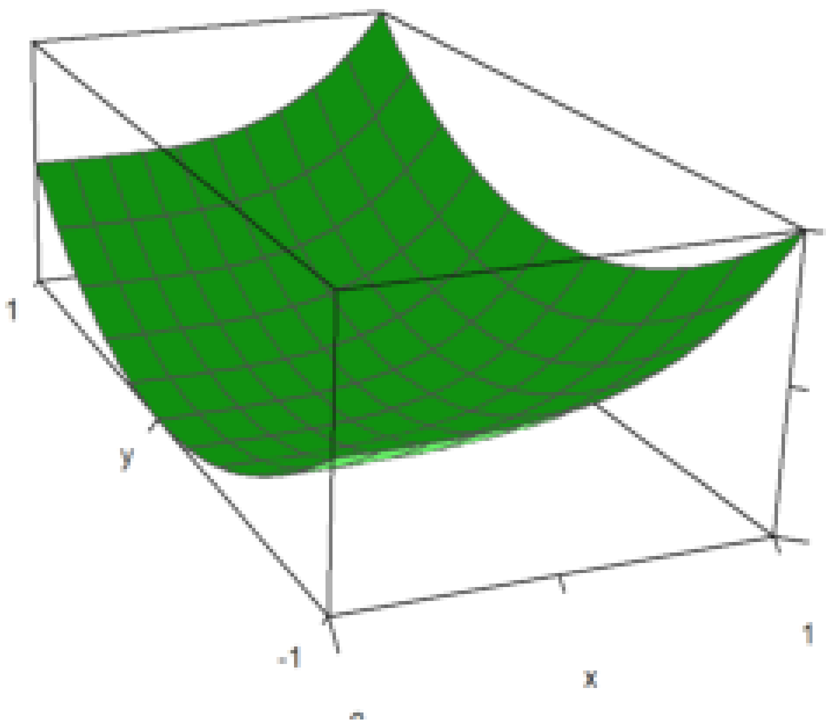
ada contoh berikut, sudut=0 dan tinggi=0 dilihat dari sumbu y negatif. Label sumbu untuk y disembunyikan dalam kasus ini.

```
>plot3d("x^2+y",distance=3,zoom=1,angle=pi/2,height=0):
```



Plot selalu terlihat berada di tengah kubus plot. Anda dapat memindahkan bagian tengah dengan parameter tengah.

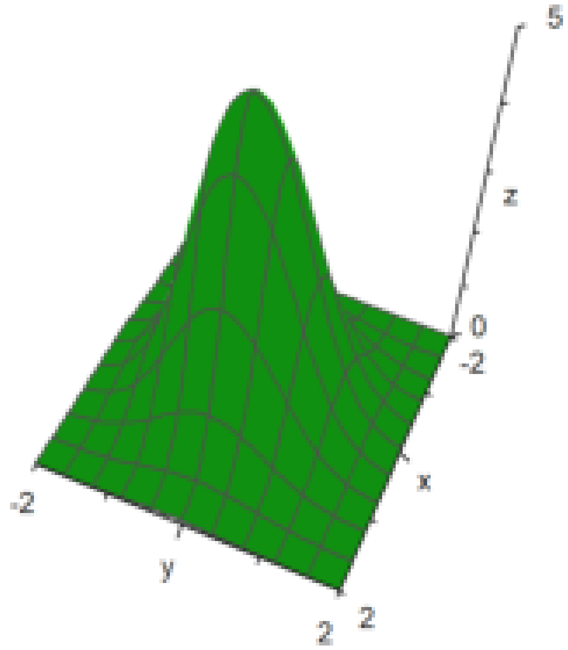
```
>plot3d("x^4+y^2",a=0,b=1,c=-1,d=1,angle=-20°,height=20°, ...
>  center=[0.4,0,0],zoom=5):
```



Plotnya diskalakan agar sesuai dengan unit kubus untuk dilihat. Jadi tidak perlu mengubah jarak atau zoom tergantung ukuran plot. Namun labelnya mengacu pada ukuran sebenarnya.

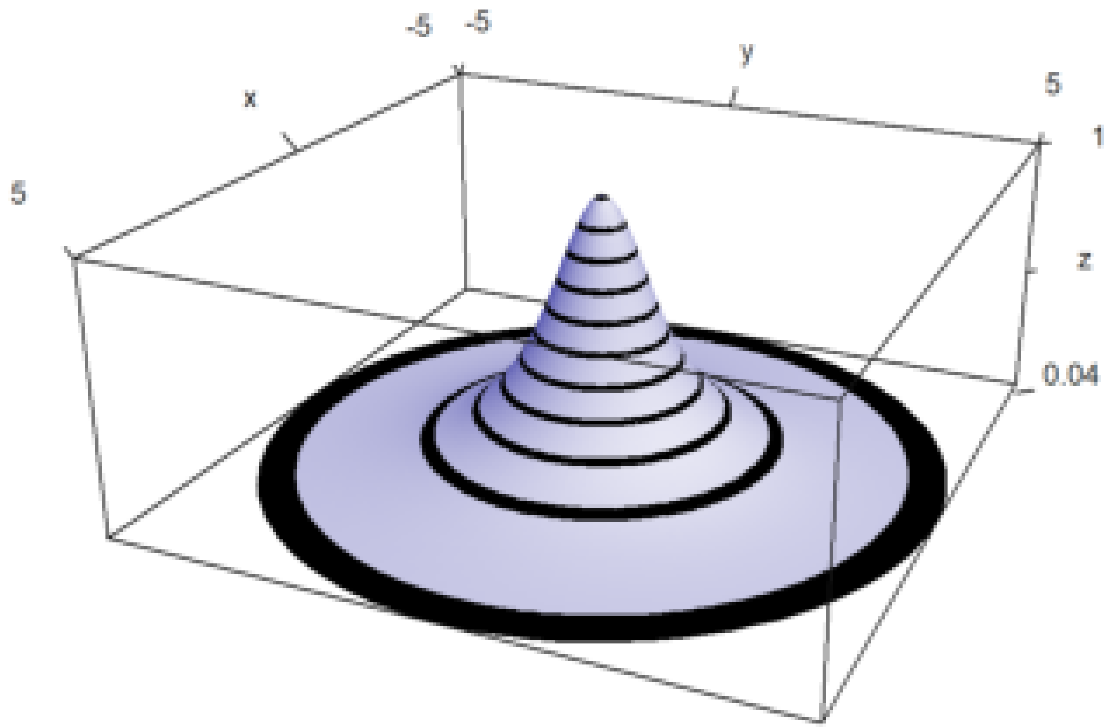
Jika Anda mematikannya dengan scale=false, Anda harus berhati-hati agar plot tetap masuk ke dalam jendela plotting, dengan mengubah jarak pandang atau zoom, dan memindahkan bagian tengah.

```
>plot3d("5*exp(-x^2-y^2)",r=2,<fscale,<scale,distance=13,height=50°, ...
>  center=[0,0,-2],frame=3):
```

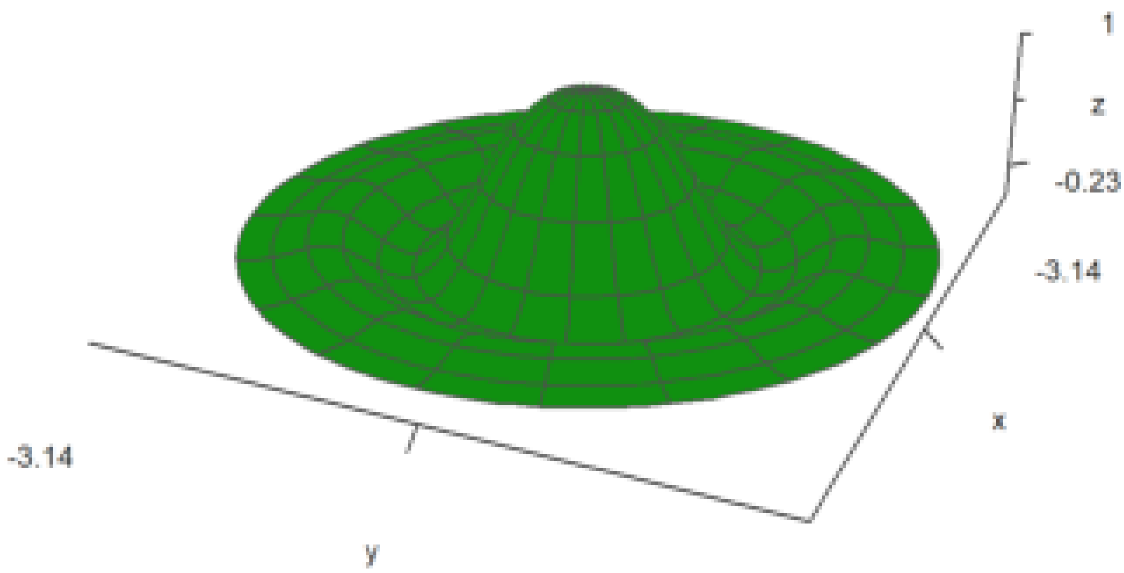


Plot kutub juga tersedia. Parameter polar=true menggambar plot kutub. Fungsi tersebut harus tetap merupakan fungsi dari x dan y. Parameter "fscale" menskalakan fungsi dengan skalanya sendiri. Kalau tidak, fungsinya akan diskalakan agar sesuai dengan kubus.

```
>plot3d("1/(x^2+y^2+1)",r=5,>polar, ...
>fscale=2,>hue,n=100,zoom=4,>contour,color=blue):
```



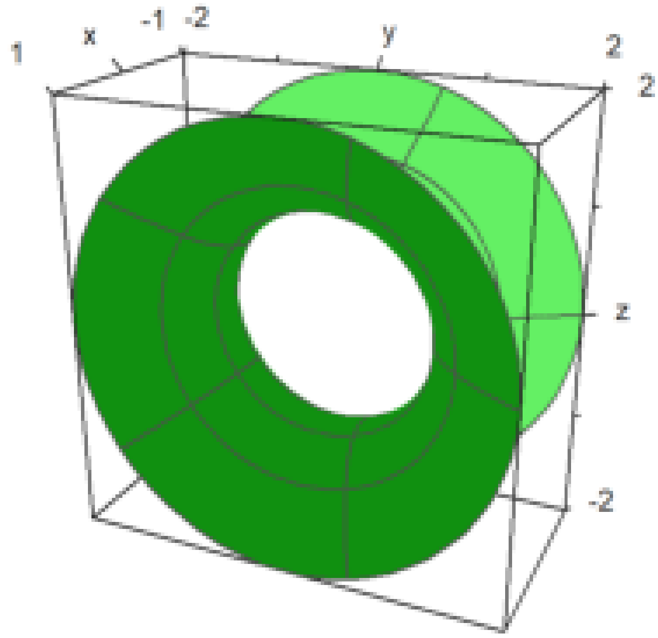

```
>function f(r) := exp(-r/2)*cos(r); ...
>plot3d("f(x^2+y^2)",>polar,scale=[1,1,0.4],r=pi,frame=3,zoom=4):
```



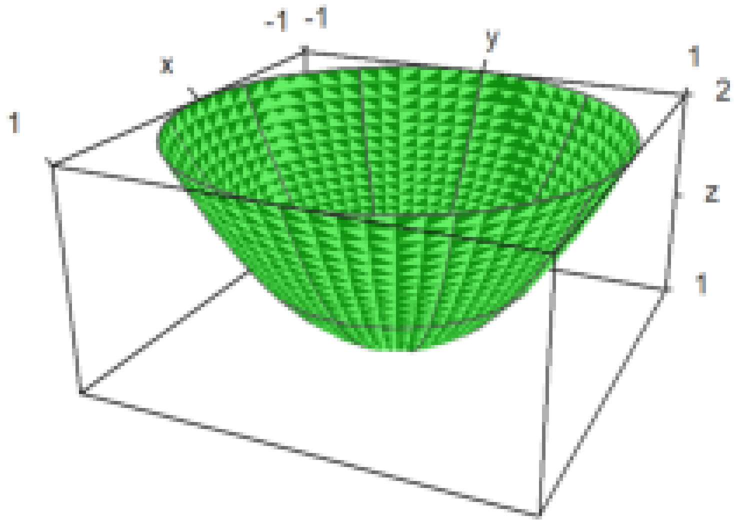
Parameter memutar memutar fungsi di x di sekitar sumbu x.

- rotate 1: Menggunakan sumbu x
- rotate 2: Menggunakan sumbu z

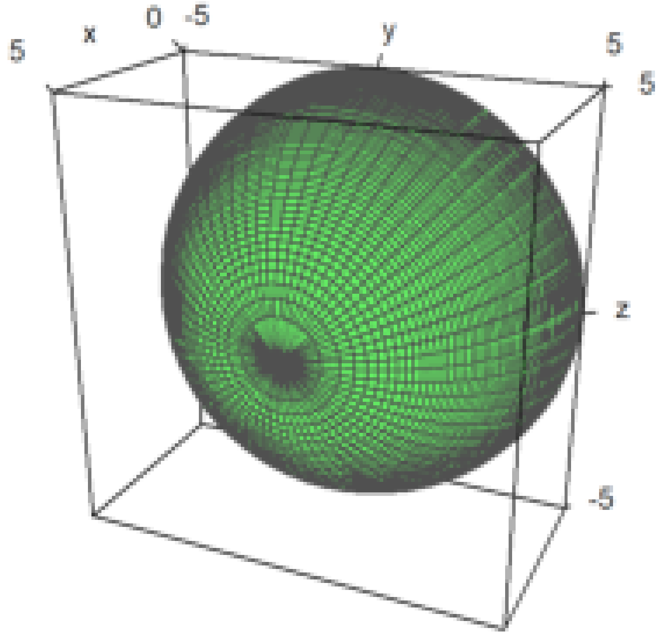
```
>plot3d("x^2+1",a=-1,b=1,rotate=true,grid=5):
```



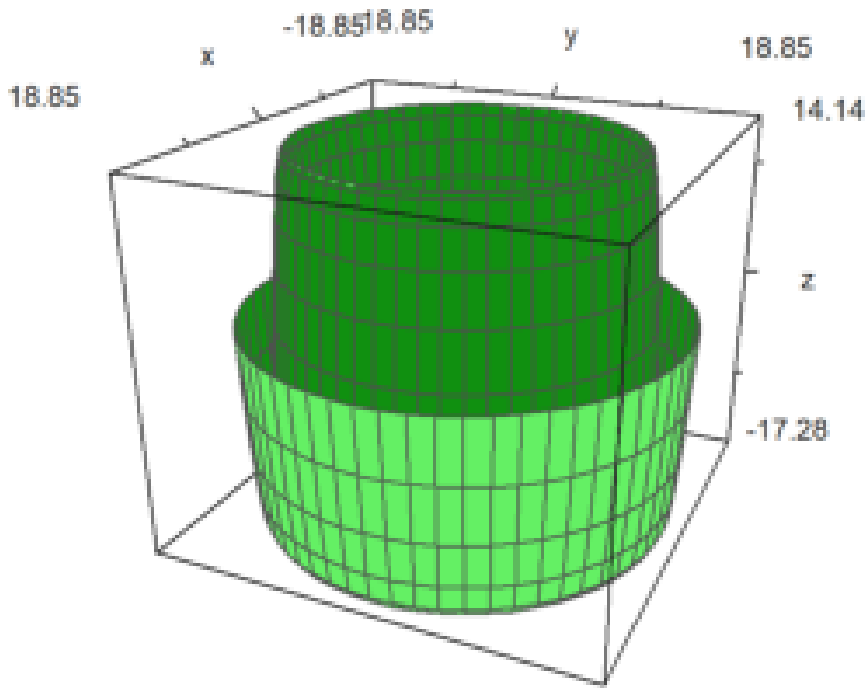

```
>plot3d("x^2+1",a=-1,b=1,rotate=2,grid=5):
```




```
>plot3d("sqrt(25-x^2)",a=0,b=5,rotate=1):
```

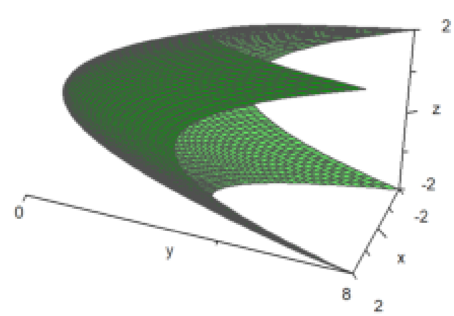



```
>plot3d("x*sin(x)",a=0,b=6pi,rotate=2):
```



Berikut adalah plot dengan tiga fungsi.

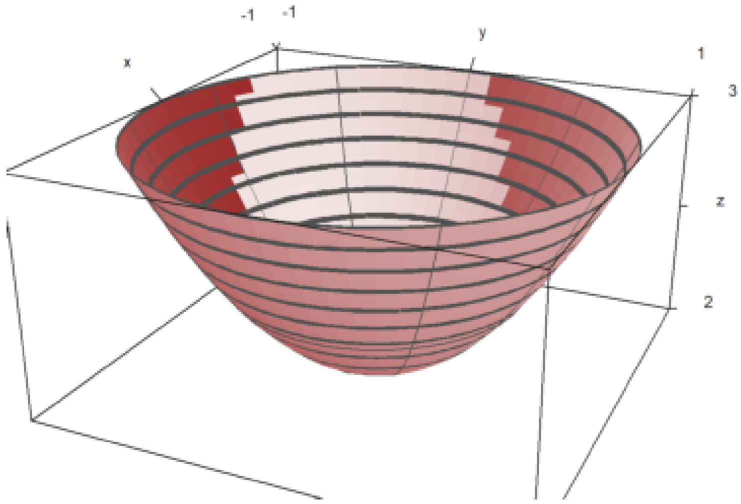
```
>plot3d("x","x^2+y^2","y",r=2,zoom=3.5,frame=3):
```



Contoh Soal:
1. Gambarlah grafik dari fungsi berikut.

$$f(x,y) = x^2 + 4y^2$$

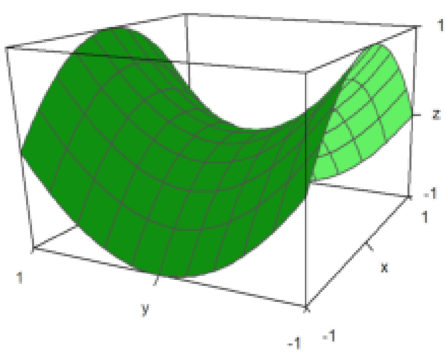
```
>plot3d("x^2+2",a=-1,b=1,rotate=2,grid=5,zoom=5,>polar,>hue,>contour,color=
```



2. Gambarlah grafik dari fungsi berikut.

$$f(x,y) = y^2 - x^2$$

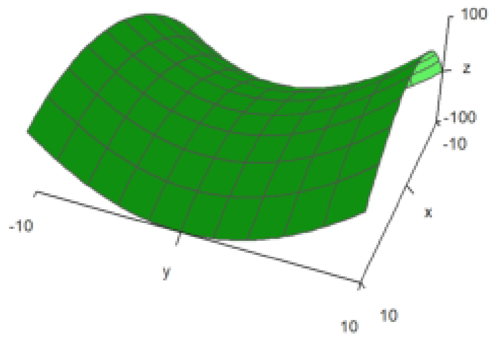
```
>plot3d("y^2-x^2",a=-1,b=1,angle=-20,center=[1,1,0],zoom=3,distance=4,heigh
```



3. Gambarlah grafik dari fungsi berikut.

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

```
>function f(x,y):=y^2-x^2;
>plot3d("f",r=10,n=80,fscale=4,scale=1.2,frame=3,>user):
```



Plot Kontur

Untuk plotnya, Euler menambahkan garis grid. Sebaliknya dimungkinkan untuk menggunakan garis datar dan rona satu warna atau rona warna spektral. Euler dapat menggambar ketinggian fungsi pada plot dengan arsiran. Di semua plot 3D, Euler dapat menghasilkan anaglyph.

- >hue: Mengaktifkan bayangan cahaya, bukan kabel.

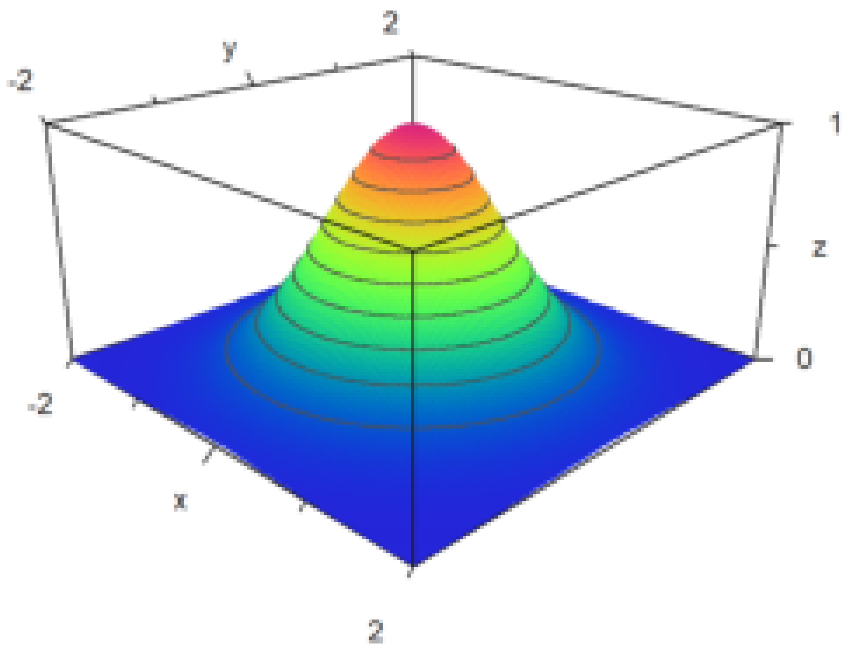
```
>contour: : Membuat plot garis kontur otomatis pada plot.
```

- level=... (or levels): A Vektor nilai garis kontur.

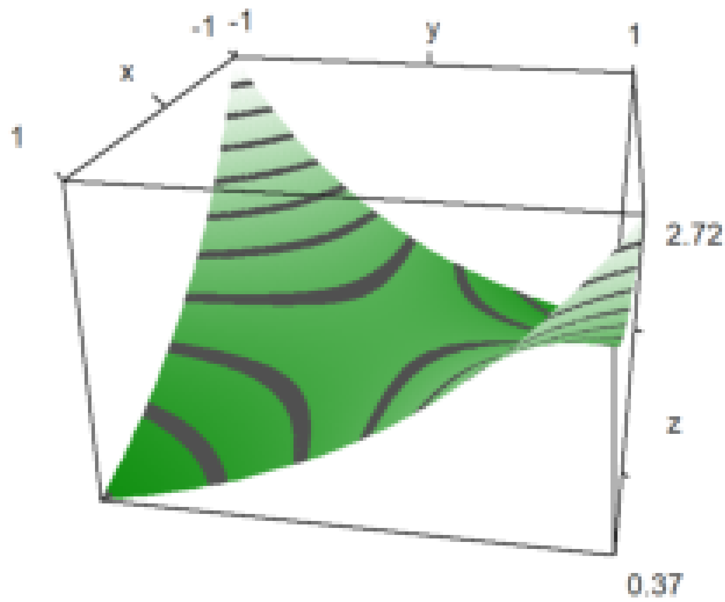
Standarnya adalah level="auto", yang menghitung beberapa garis level secara otomatis. Seperti yang Anda lihat di plot, level sebenarnya adalah rentang level.

Gaya default dapat diubah. Untuk plot kontur berikut, kami menggunakan grid yang lebih halus berukuran 100x100 poin, menskalakan fungsi dan plot, dan menggunakan sudut pandang yang berbeda.

```
>plot3d("exp(-x^2-y^2)",r=2,n=100,level="thin", ...
> >contour,>spectral,fscale=1,scale=1.1,angle=45°,height=20°):
```



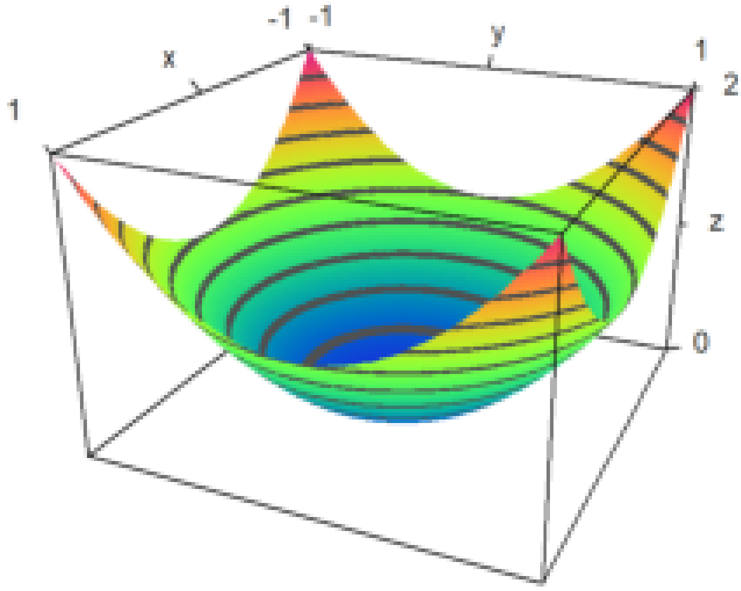

```
>plot3d("exp(x*y)",angle=100°,>contour,color=green):
```



Bayangan defaultnya menggunakan warna abu-abu. Namun rentang warna spektral juga tersedia.
- >spectral: Menggunakan skema spektral default
- color=...: Menggunakan warna khusus atau skema

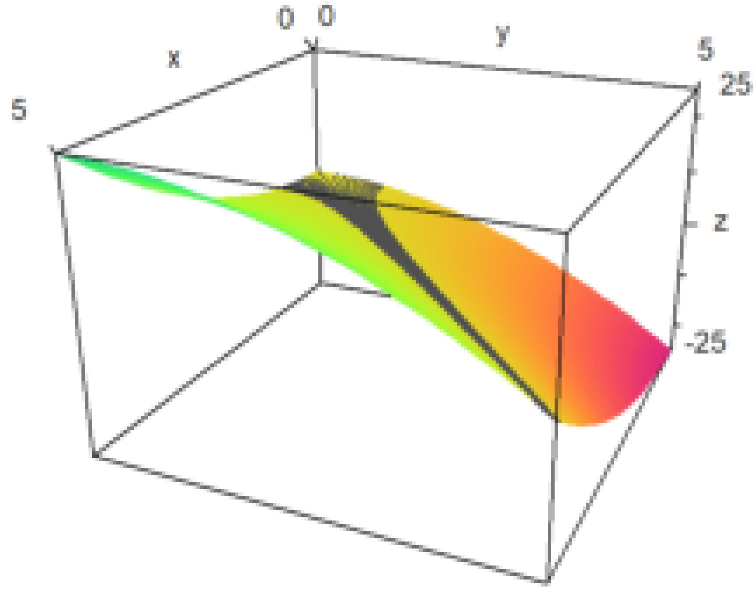
spektral Untuk plot berikut, kami menggunakan skema spektral default dan menambah jumlah titik untuk mendapatkan tampilan yang sangat halus.

```
>plot3d("x^2+y^2",>spectral,>contour,n=100):
```



Selain garis level otomatis, kita juga dapat menetapkan nilai garis level. Ini akan menghasilkan garis level yang tipis, bukan rentang level.

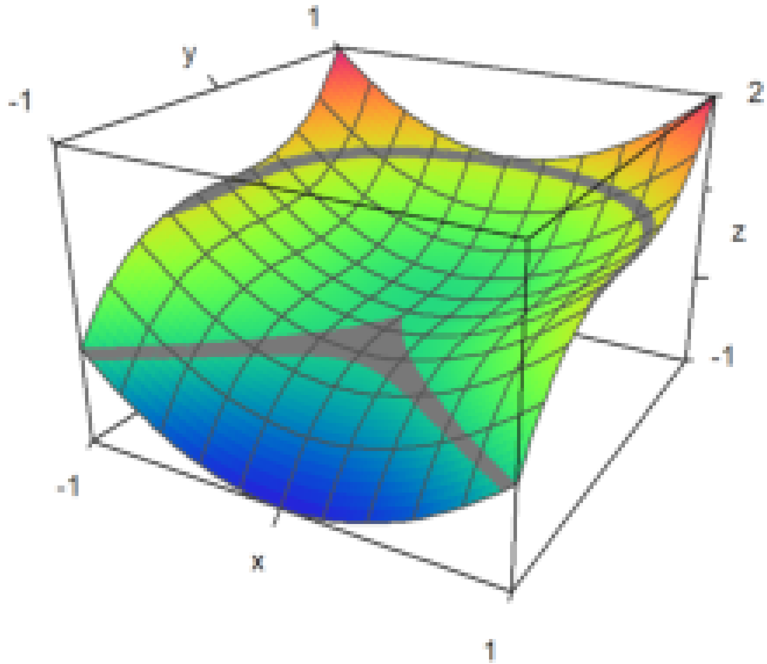
```
>plot3d("x^2-y^2",0,5,0,5,level=-1:0.1:1,color=redgreen):
```



Dalam plot berikut, kita menggunakan dua pita tingkat yang sangat luas dari -0,1 hingga 1, dan dari 0,9 hingga 1. Ini dimasukkan sebagai matriks dengan batas tingkat sebagai kolom.

Selain itu, kami melapisi grid dengan 10 interval di setiap arah.

```
>plot3d("x^2+y^3",level=[-0.1,0.9;0,1], ...
>  >spectral,angle=30°,grid=10,contourcolor=gray):
```



Pada contoh berikut, kita memplot himpunan, di mana :

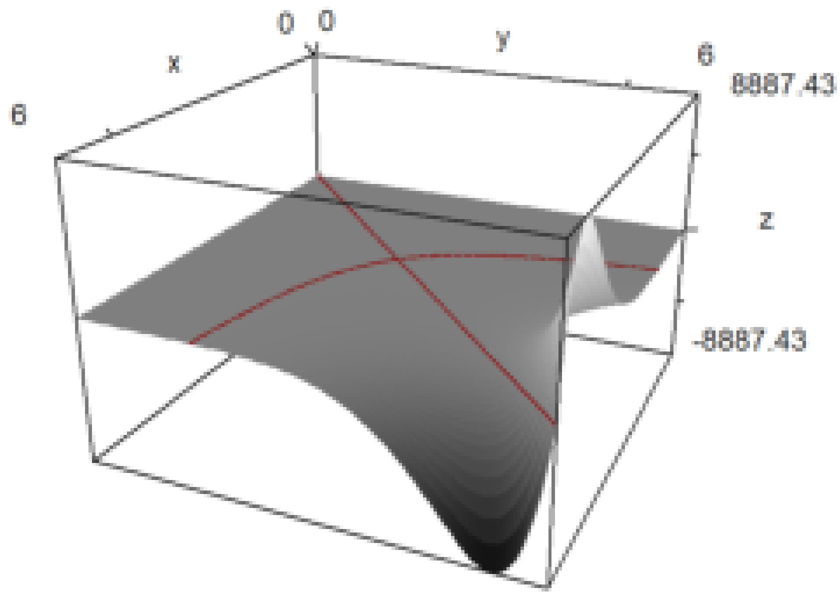
$$f(x, y) = x^y - y^x = 0$$

Kami menggunakan satu garis tipis untuk garis level.

```
>plot3d("x^y-y^x",level=0,a=0,b=6,c=0,d=6,contourcolor=red,n=100):
```

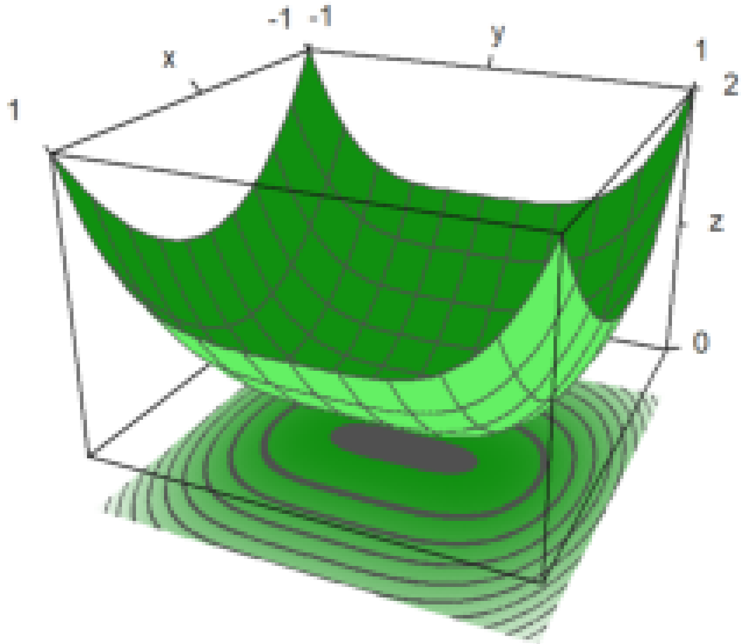
Dimungkinkan untuk menampilkan bidang kontur di bawah plot. Warna dan jarak ke plot dapat ditentukan.

```
>plot3d("x^2+y^4",>cp,cpcolor=green,cpdelta=0.2):
```



Berikut beberapa gaya lainnya. Kami selalu mematikan bingkai, dan menggunakan berbagai skema warna untuk plot dan kisi.

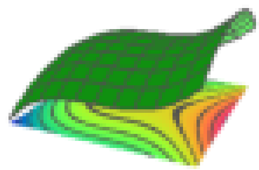
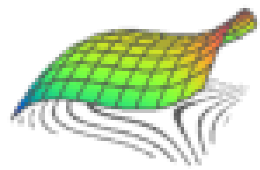
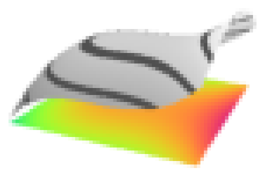
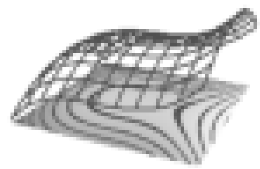
```
>figure(2,2); ...
>expr="y^3-x^2"; ...
>figure(1);  ...
>  plot3d(expr,<frame,>cp,cpcolor=spectral); ...
>figure(2);  ...
>  plot3d(expr,<frame,>spectral,grid=10,cp=2); ...
>figure(3);  ...
>  plot3d(expr,<frame,>contour,color=gray,nc=5,cp=3,cpcolor=greenred); ...
>figure(4);  ...
>  plot3d(expr,<frame,>hue,grid=10,>transparent,>cp,cpcolor=gray); ...
>figure(0):
```



Ada beberapa skema spektral lainnya, yang diberi nomor dari 1 hingga 9. Namun Anda juga dapat menggunakan warna=nilai, di mana nilai :

- spectral: untuk rentang dari biru ke merah
- white: untuk rentang yang lebih redup
- yellowblue,purplegreen,blueyellow,greenred
- blueyellow, greenpurple,yellowblue,redgreen

```
>figure(3,3); ...
>for i=1:9;  ...
>  figure(i); plot3d("x^2+y^2",spectral=i,>contour,>cp,<frame,zoom=4);  ...
>end; ...
>figure(0):
```

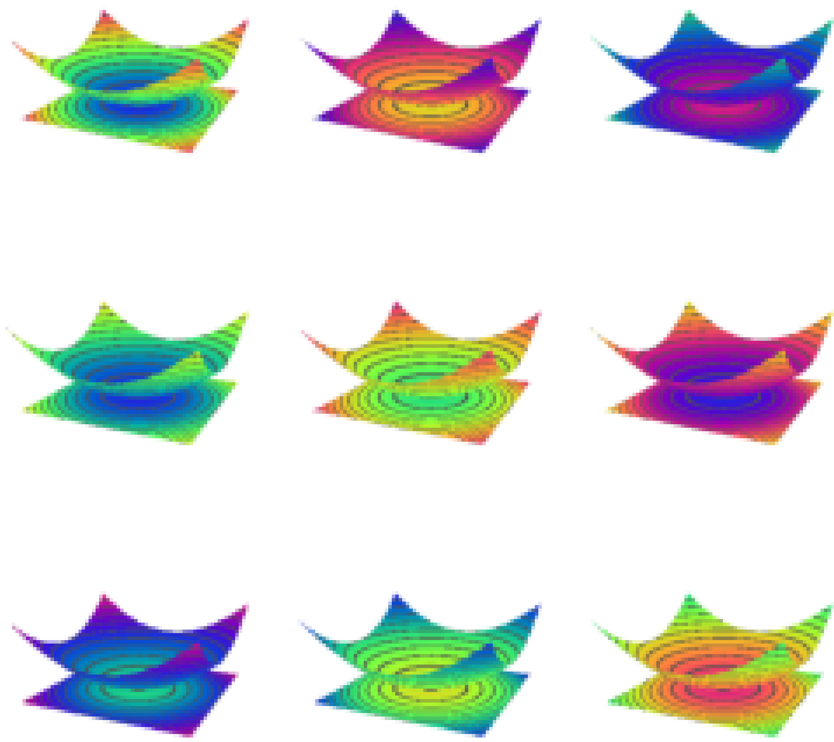





Sumber cahaya dapat diubah dengan l dan tombol kursor selama interaksi pengguna. Itu juga dapat diatur dengan parameter.

- light: arah
- amb: cahaya sekitar antara 0 dan 1

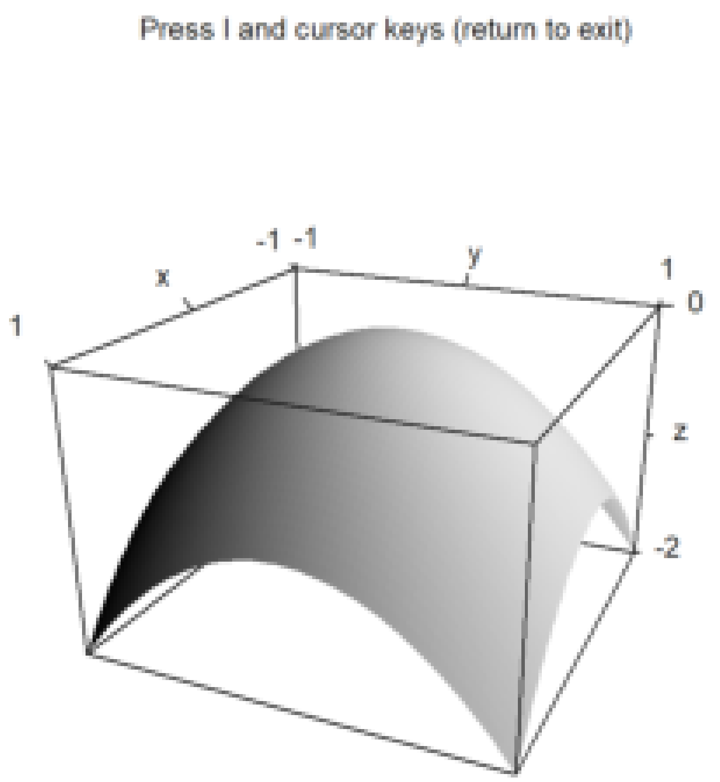
Catatan : program tidak membuat perbedaan antara sisi plot. Tidak ada bayangan. Untuk ini, Anda memerlukan Povray.

```
>plot3d("-x^2-y^2", ...
>  hue=true,light=[0,1,1],amb=0,user=true, ...
>  title="Press l and cursor keys (return to exit)"):
```



Parameter warna mengubah warna permukaan. Warna garis level juga bisa diubah.

```
>plot3d("-x^2-y^2",color=rgb(0.2,0.2,0),hue=true,frame=false, ...
>  zoom=3,contourcolor=red,level=-2:0.1:1,dl=0.01):
```

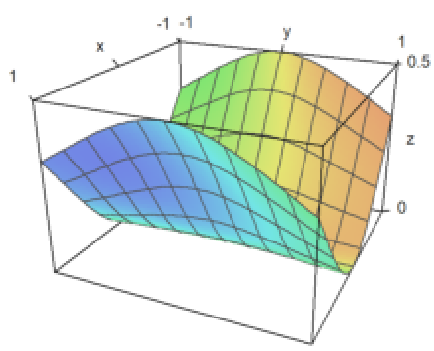


The color 0 gives a special rainbow effect.

```
>plot3d("x^2/(x^2+y^2+1)",color=0,hue=true,grid=10):
```

Permukaannya juga bisa transparan.

```
>plot3d("x^2+y^2",>transparent,grid=10,wirecolor=red):
```

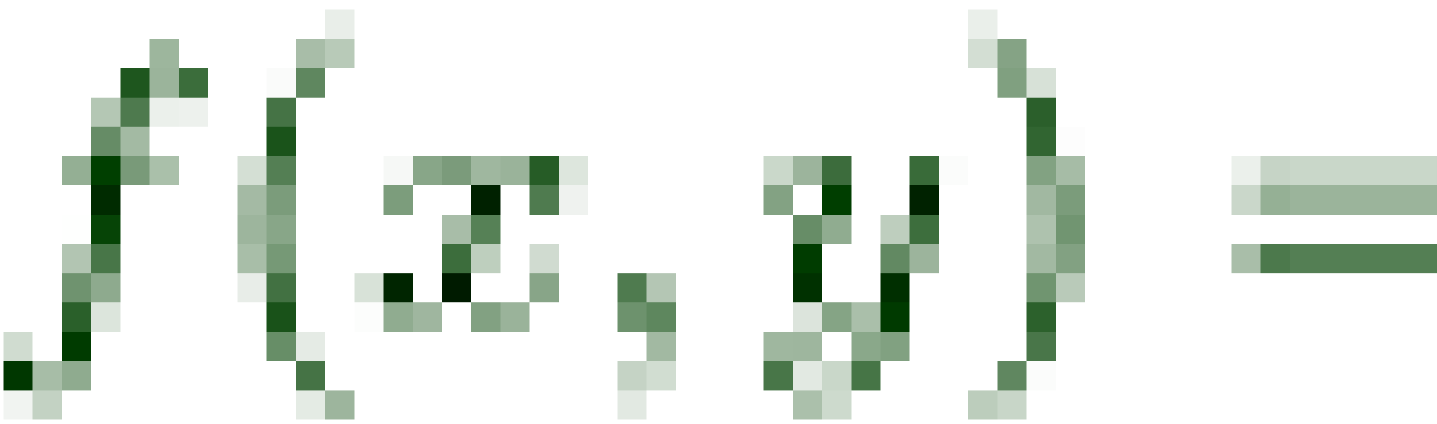


Contoh Soal:

1. Gambarlah plot kontur dari fungsi berikut.

$$f(x, y) = e^{x^2-y^2}$$

```
>plot3d("exp(x^2-y^2)",r=2,n=100,level="thin", ...
>>contour,>spectral,fscale=2,scale=1.1,angle=45°,height=20°,zoom=2.5):
```



2. Gambarlah plot kontur dari fungsi berikut.

$$f(x, y) = -x^3 - y^2$$

```
>plot3d("-x^3-y^2",color=rgb(0.1,0.2,0),hue=true,frame=false, ...
>zoom=4,contourcolor=black,level=-2:0.1:1,dl=0.01):
```

$$f(x,y)$$

3. Gambarlah plot kontur dari fungsi berikut.

$$f(x,y)=\frac{y}{x^2+y^2+1}$$

```
>plot3d("y/(x^2+y^2+1)",color=0,hue=true,grid=0,zoom=3.5):
```

$$f(x,y)=\frac{}{x^2}$$

4. Gambarlah plot kontur transparan dari fungsi berikut.

$$1+\cos\left(\frac{y}{1+x^2+y^2}\right)$$

```
>plot3d("1+cos(y/1+x^2+y^2)",>transparent,grid=10,wirecolor=green):
```

$$1+\cos\left(\frac{\quad}{1+\cdots}\right.$$

Plot Implisit

Ada juga plot implisit dalam tiga dimensi. Euler menghasilkan pemotongan melalui objek. Fitur plot3d mencakup plot implisit. Plot ini menunjukkan himpunan nol suatu fungsi dalam tiga variabel. Permukaannya juga bisa transparan.
Solusi dari

atex: f(x,y,z) = 0

dapat divisualisasikan dalam potongan yang sejajar dengan bidang xy-, xz- dan yz.

- implicit=1: memotong sejajar dengan bidang y-z
- implicit=2: memotong sejajar dengan bidang x-z
- implicit=4: memotong sejajar dengan bidang x-y

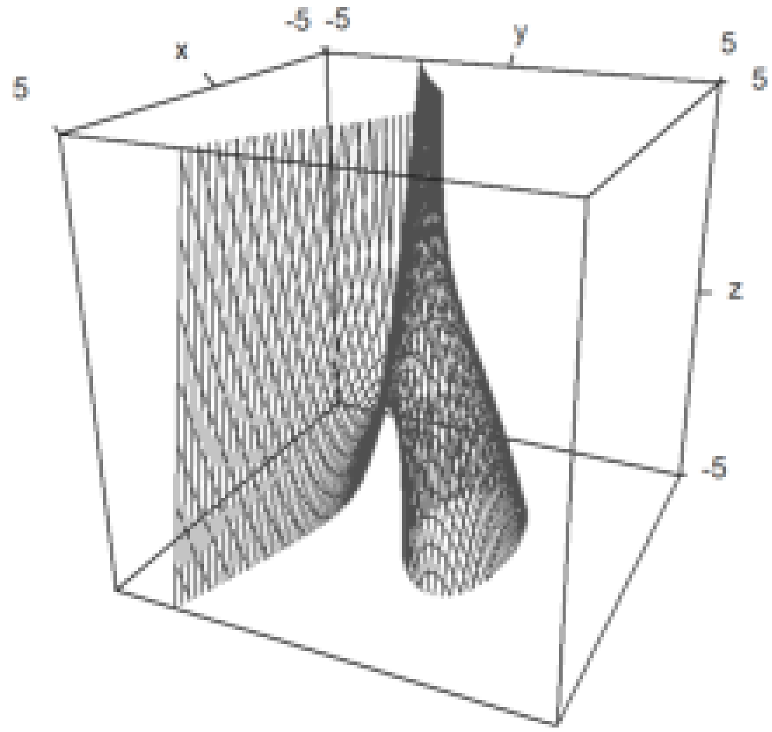
Tambahkan nilai berikut, jika Anda mau. Dalam contoh kita memplot :

$$M=\{(x,y,z):x^2+y^3+zy=1\}$$

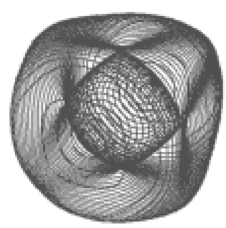
```
>plot3d("x^2+y^3+z*y-1",r=5,implicit=3):
```

$$N =$$

```
>c=1; d=1;
>plot3d("((x^2+y^2-c^2)^2+(z^2-1)^2)*((y^2+z^2-c^2)^2+(x^2-1)^2)*((z^2+x^2-
```




```
>plot3d("x^2+y^2+4*x*z+z^3",>implicit,r=2,zoom=2.5):
```



Contoh Soal:
Gambarlah grafik dari fungsi implisit berikut.

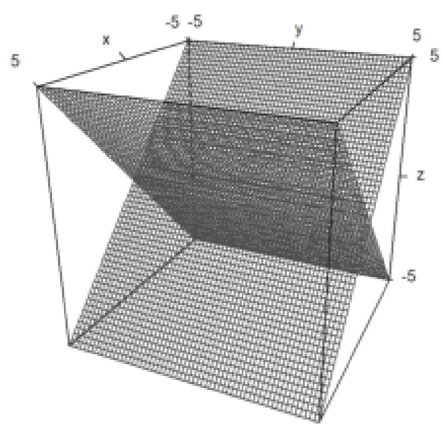
$$f(x, y) = x^2 - z^2$$

$$f(x, y) = 9x^2 + 4z^2 - 36 * y$$

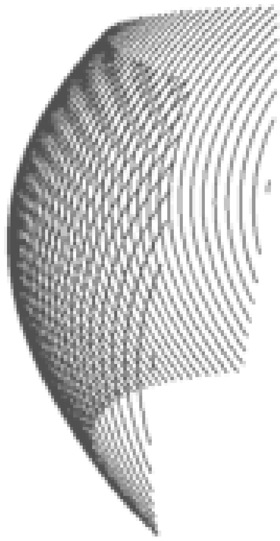
$$f(x, y) = y^2 + z^2 - 12y$$

```
>plot3d("x^2-z^2",r=5,implicit=3):
```


```
>plot3d("9*x^2+4*z^2-36*y",r=4,<frame,>implicit,>user):
```



```
>plot3d("y^2+z^2-12*y",>implicit,r=2,zoom=2.5):
```



Merencanakan Data 3D

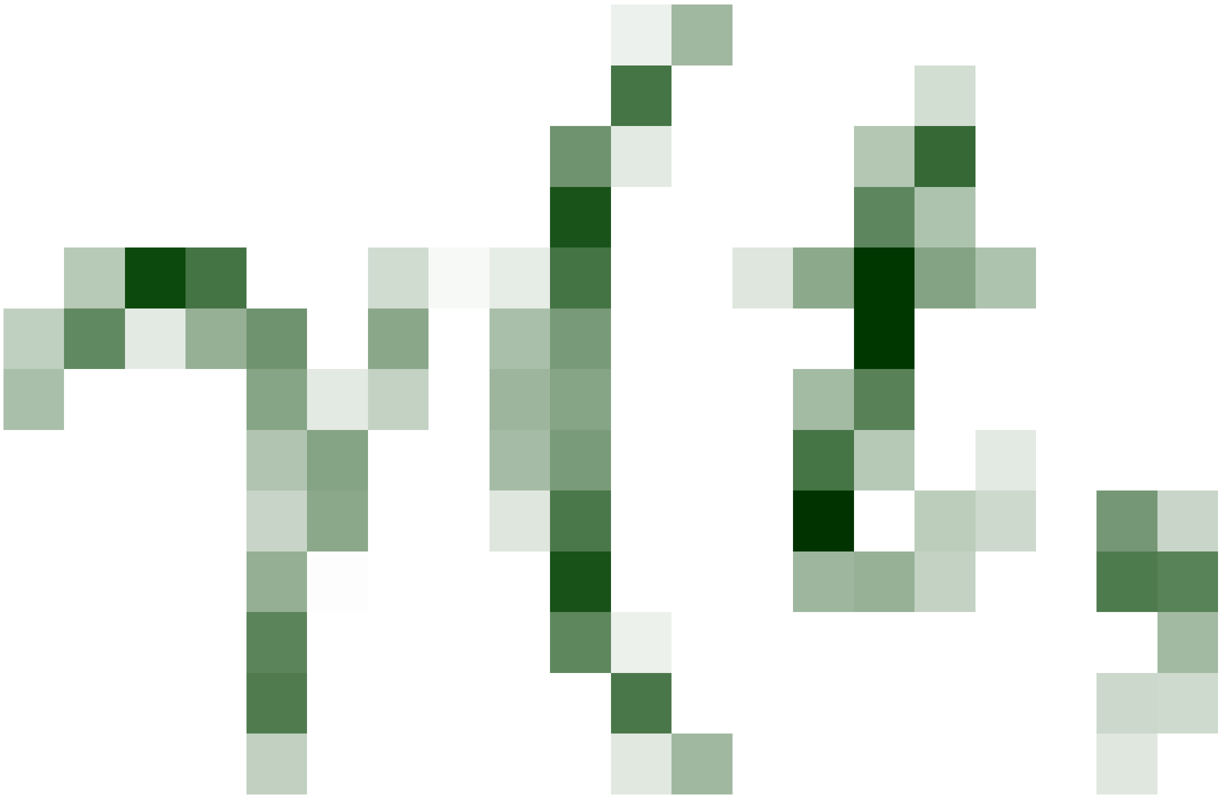
Sama seperti plot2d, plot3d menerima data. Untuk objek 3D, Anda perlu menyediakan matriks nilai x-, y- dan z, atau tiga fungsi atau ekspresi fx(x,y), fy(x,y), fz(x,y).

$$\gamma(t, s) = (x(t, s), y(t, s), z(t, s))$$

Karena x,y,z adalah matriks, kita asumsikan bahwa (t,s) melewati grid persegi. Hasilnya, Anda dapat memplot gambar persegi panjang di ruang angkasa.
Anda dapat menggunakan bahasa matriks Euler untuk menghasilkan koordinat secara efektif.

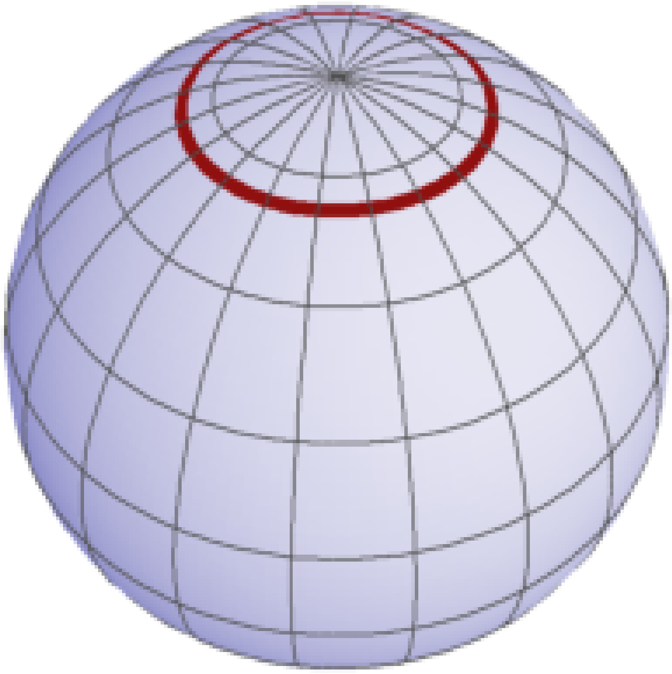
Dalam contoh berikut, kita menggunakan vektor nilai t dan vektor kolom nilai s untuk membuat parameter permukaan bola. Dalam gambar kita dapat menandai wilayah, dalam kasus kita wilayah kutub.

```
>t=linspace(0,2pi,180); s=linspace(-pi/2,pi/2,90)'; ...
>x=cos(s)*cos(t); y=cos(s)*sin(t); z=sin(s); ...
>plot3d(x,y,z,>hue, ...
>color=blue,<frame,grid=[10,20], ...
>values=s,contourcolor=red,level=[90°-24°;90°-22°], ...
>scale=1.4,height=50°):
```



Berikut adalah contoh, yang merupakan grafik dari sebuah fungsi.

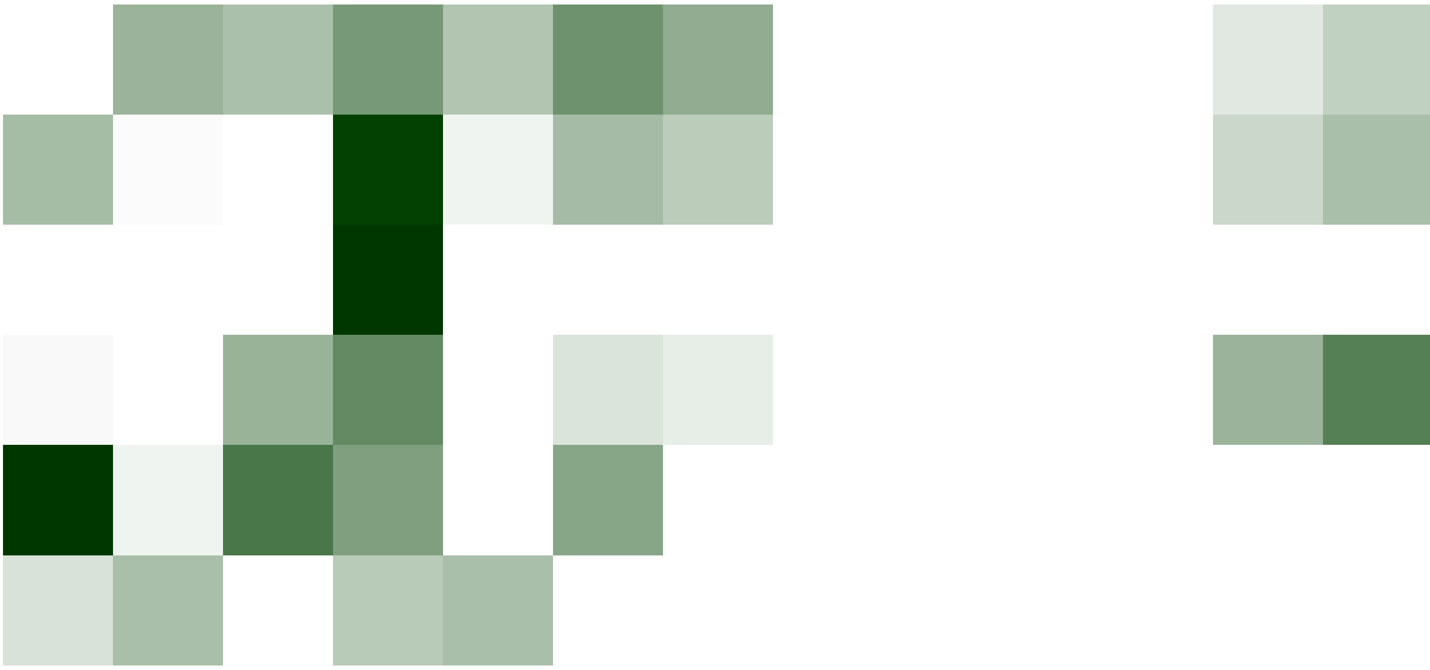
```
>t=-1:0.1:1; s=(-1:0.1:1)'; plot3d(t,s,t*s,grid=10):
```



Namun, kita bisa membuat berbagai macam permukaan. Berikut adalah permukaan yang sama sebagai suatu fungsi :

$$x = y\,z$$

```
>plot3d(t*s,t,s,angle=180°,grid=10):
```



Dengan lebih banyak usaha, kita dapat menghasilkan banyak permukaan.

Dalam contoh berikut kita membuat tampilan bayangan dari bola yang terdistorsi. Koordinat bola yang biasa adalah

$$\gamma(t,s) = (\cos(t)\cos(s), \sin(t)\sin(s), \cos(s))$$

dengan

$$0 \leq t \leq 2\pi, \quad \frac{-\pi}{2} \leq s \leq \frac{\pi}{2}.$$

Kami mendistorsi ini dengan sebuah faktor

$$d(t,s) = \frac{\cos(4t) + \cos(8s)}{4}.$$

```
>t=linspace(0,2pi,320); s=linspace(-pi/2,pi/2,160)'; ...
>d=1+0.2*(cos(4*t)+cos(8*s)); ...
>plot3d(cos(t)*cos(s)*d,sin(t)*cos(s)*d,sin(s)*d,hue=1, ...
>  light=[1,0,1],frame=0,zoom=5):
```

$$d(t, s) =$$

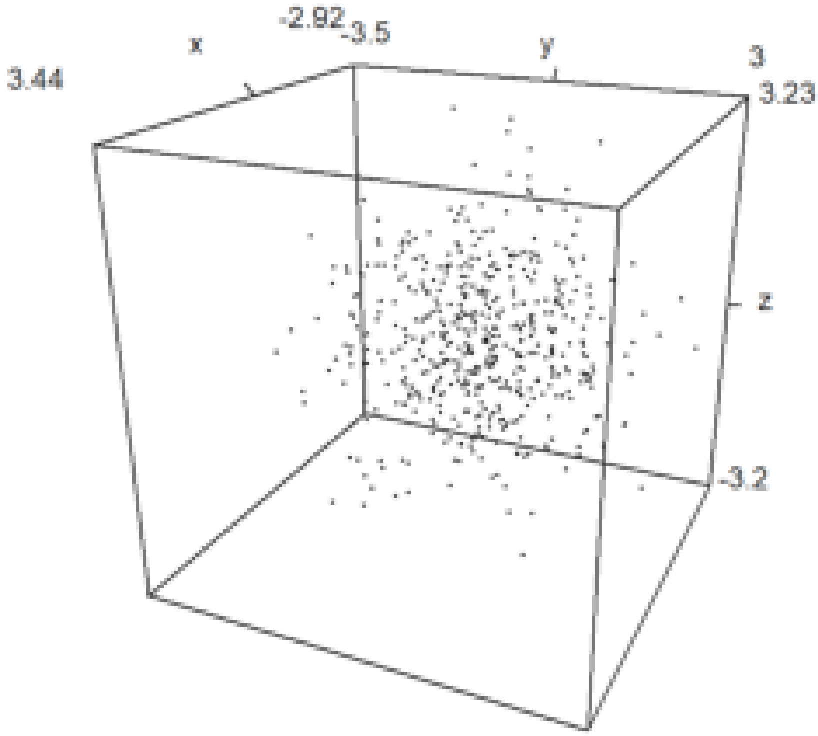
Tentu saja, point cloud juga dimungkinkan. Untuk memplot data titik dalam ruang, kita memerlukan tiga vektor untuk koordinat titik-titik tersebut.

Gayanya sama seperti di plot2d dengan points=true;

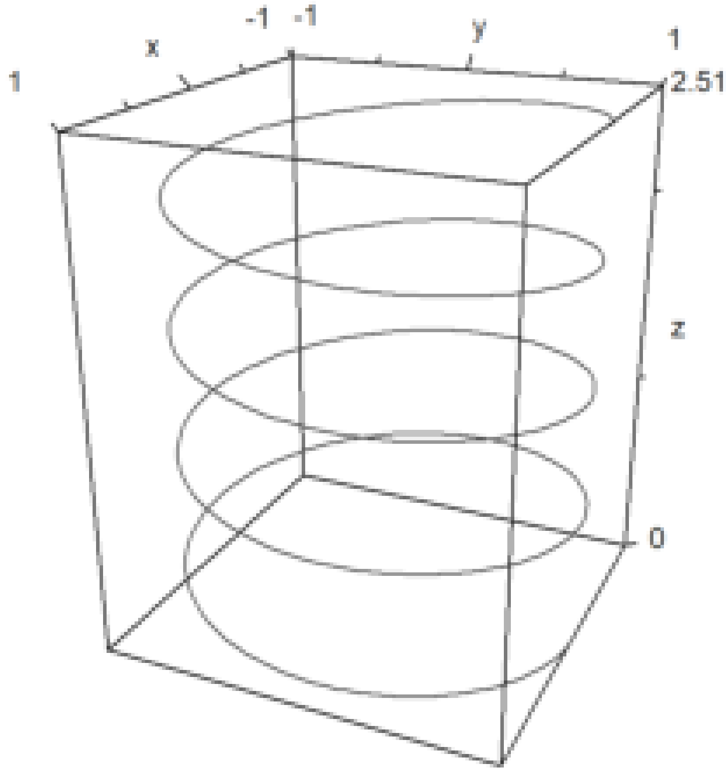
```
>n=500;  ...
>  plot3d(normal(1,n),normal(1,n),normal(1,n),points=true,style="."):
```

Dimungkinkan juga untuk memplot kurva dalam 3D. Dalam hal ini, lebih mudah untuk menghitung terlebih dahulu titik-titik kurva. Untuk kurva pada bidang kita menggunakan barisan koordinat dan parameter wire=true.

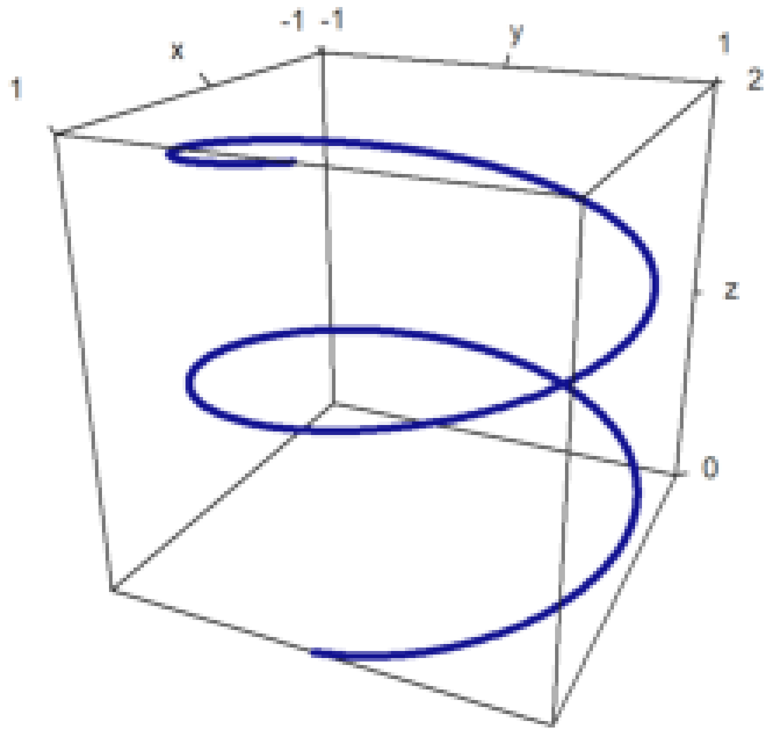
```
>t=linspace(0,8pi,500); ...
>plot3d(sin(t),cos(t),t/10,>wire,zoom=3):
```




```
>t=linspace(0,4pi,1000); plot3d(cos(t),sin(t),t/2pi,>wire, ...
>linewidth=3,wirecolor=blue):
```

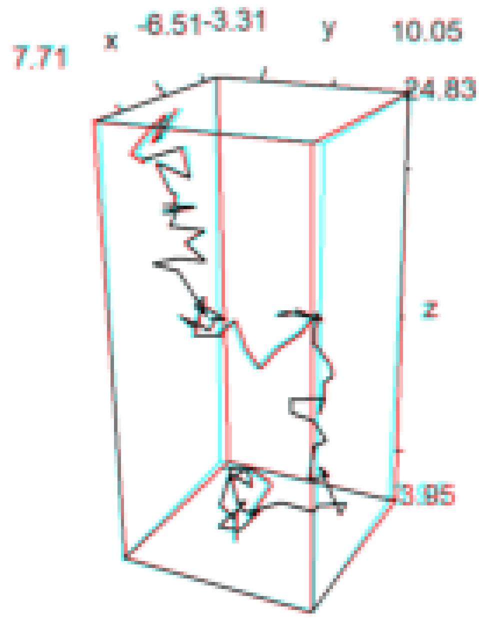



```
>X=cumsum(normal(3,100)); ...
> plot3d(X[1],X[2],X[3],>anaglyph,>wire):
```



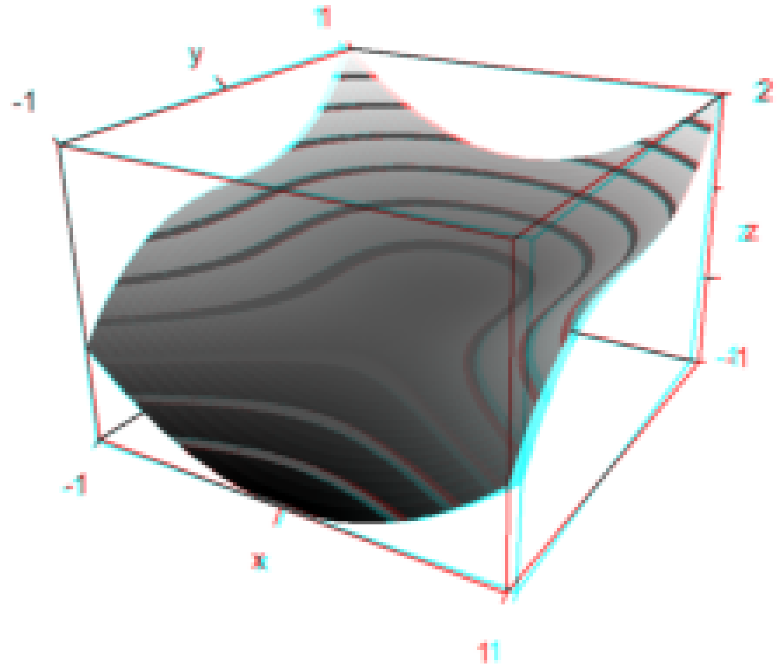
EMT juga dapat membuat plot dalam mode anaglyph. Untuk melihat plot seperti itu, Anda memerlukan kacamata berwarna merah/cyan.

```
> plot3d("x^2+y^3",>anaglyph,>contour,angle=30°):
```



Seringkali skema warna spektral digunakan untuk plot. Ini menekankan ketinggian fungsinya.

```
>plot3d("x^2*y^3-y",>spectral,>contour,zoom=3.2):
```



Euler juga dapat memplot permukaan yang diparameterisasi, jika parameternya adalah nilai x, y, dan z dari gambar kotak persegi panjang di ruang tersebut.

Untuk demo berikut, kami menyiapkan parameter u- dan v-, dan menghasilkan koordinat ruang dari parameter tersebut.

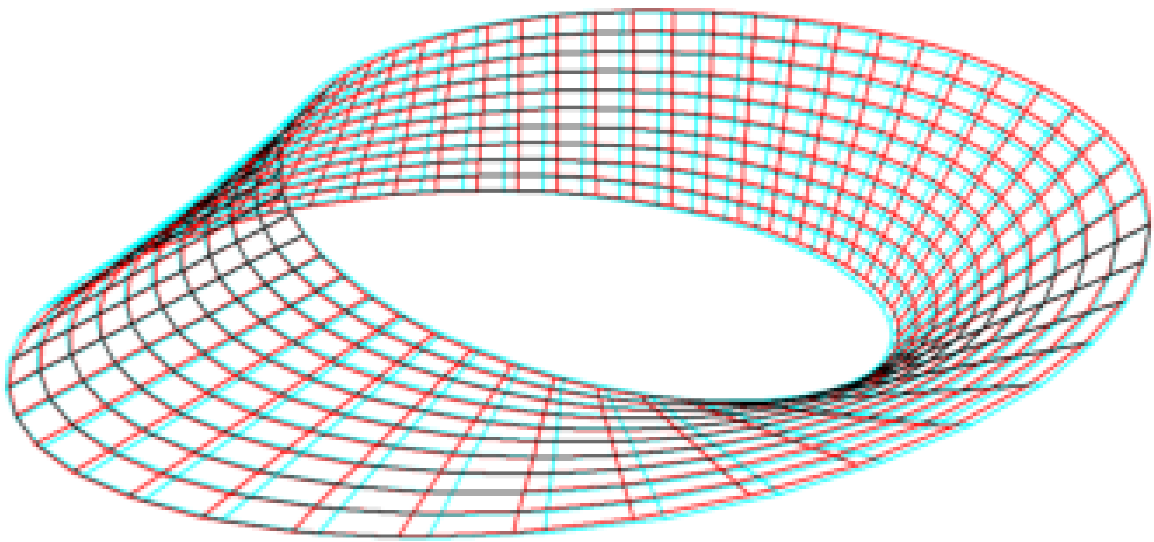
$$t \leq 2\pi, \quad \frac{-\pi}{2} \leq s \leq \frac{\pi}{2}.$$

```
>u=linspace(-1,1,10); v=linspace(0,2*pi,50)'; ...
>X=(3+u*cos(v/2))*cos(v); Y=(3+u*cos(v/2))*sin(v); Z=u*sin(v/2); ...
>plot3d(X,Y,Z,>anaglyph,<frame,>wire,scale=2.3):
```

$$t \leq 2\pi$$

Berikut adalah contoh yang lebih rumit, yang sangat megah ketika dilihat dengan kacamata merah/biru.

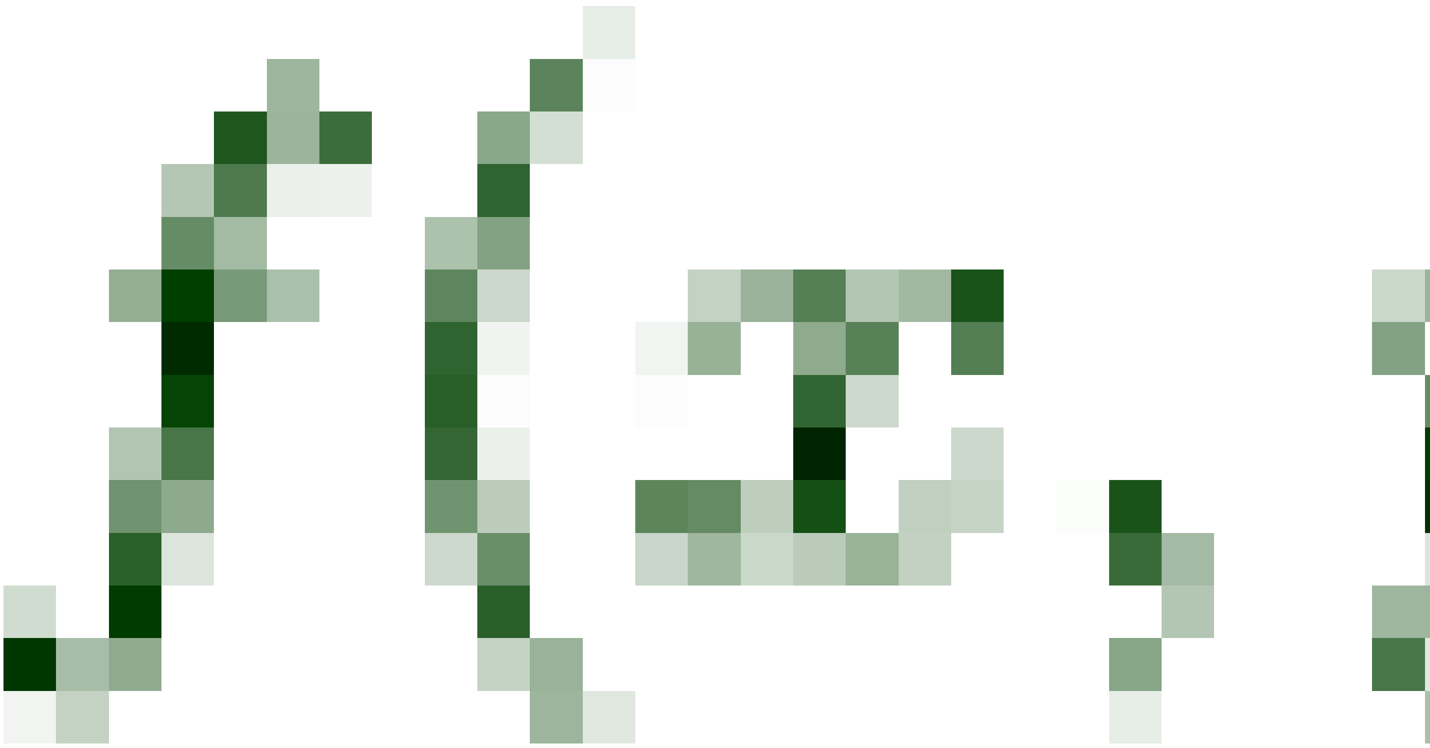
```
>u:=linspace(-pi,pi,160); v:=linspace(-pi,pi,400)';  ...
>x:=(4*(1+.25*sin(3*v))+cos(u))*cos(2*v); ...
>y:=(4*(1+.25*sin(3*v))+cos(u))*sin(2*v); ...
> z=sin(u)+2*cos(3*v); ...
>plot3d(x,y,z,frame=0,scale=1.5,hue=1,light=[1,0,-1],zoom=2.8,>anaglyph):
```



Contoh Soal:
1. Buatlah grafik anaglyph dari fungsi berikut.

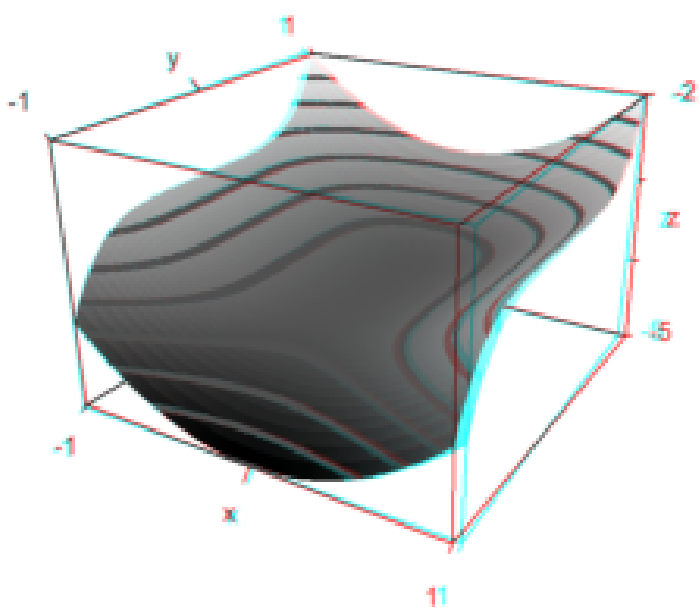
$$f(x, y) = x^2 + y^5 - 4$$

```
>plot3d("x^2+y^5-4",>anaglyph,>contour,angle=30°):
```



2. Gambarlah grafik dari data berikut.

```
>t=linspace(0,4pi,1000); plot3d(sin(t),cos(t),3t/2pi,>wire, ...
>linewidth=3,wirecolor=blue):
```



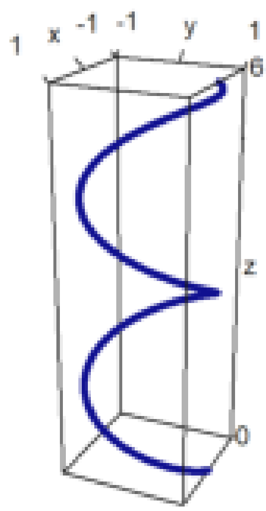
Grafik batang juga dapat dibuat. Untuk ini, kita perlu menyediakan:

- x: vektor baris dengan n+1 elemen
- y: vektor kolom dengan n+1 elemen
- z: matriks nxn dari nilai-nilai.

z bisa lebih besar, tetapi hanya nilai nxn yang akan digunakan.

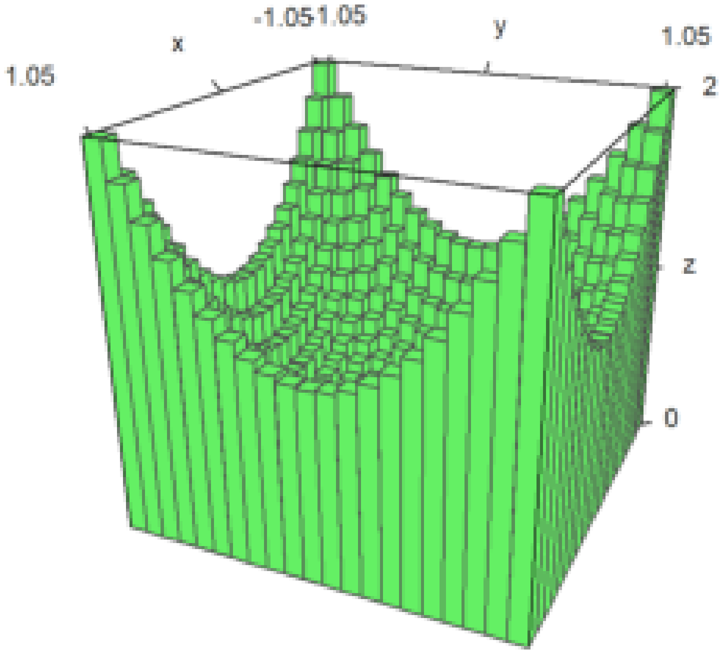
Dalam contoh ini, pertama-tama kita menghitung nilai-nilainya. Kemudian kita menyesuaikan x dan y sehingga vektor-vektornya berpusat pada nilai yang digunakan.

```
>x=-1:0.1:1; y=x'; z=x^2+y^2; ...
>xa=(x|1.1)-0.05; ya=(y_1.1)-0.05; ...
>plot3d(xa,ya,z,bar=true):
```



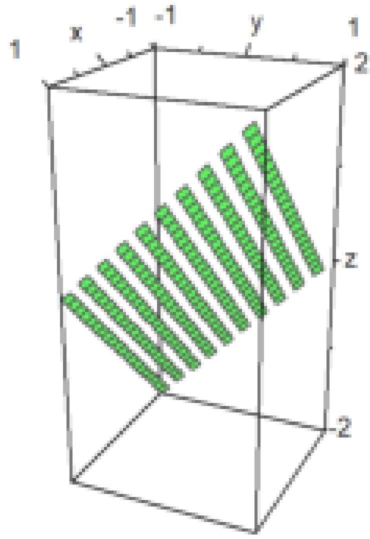
Dimungkinkan untuk membagi plot suatu permukaan menjadi dua bagian atau lebih.

```
>x=-1:0.1:1; y=x'; z=x+y; d=zeros(size(x)); ...
>plot3d(x,y,z,disconnect=2:2:20):
```

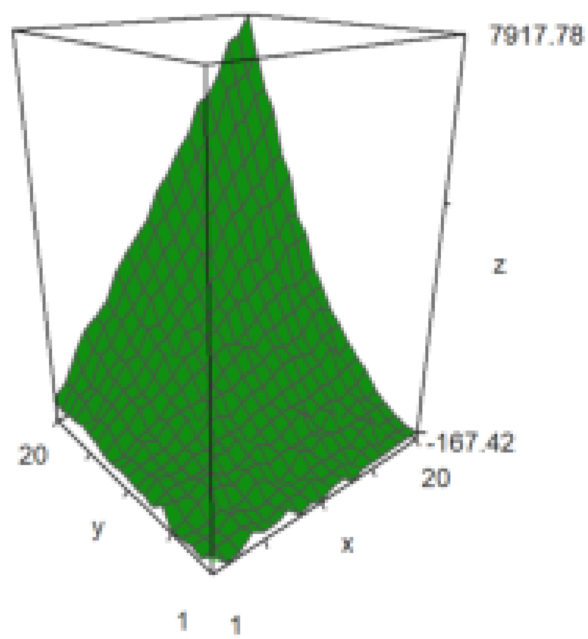


Jika memuat atau menghasilkan matriks data M dari file dan perlu memplotnya dalam 3D, Anda dapat menskalakan matriks ke [-1,1] dengan skala(M), atau menskalakan matriks dengan >zscale. Hal ini dapat dikombinasikan dengan faktor penskalaan individual yang diterapkan sebagai tambahan.

```
>i=1:20; j=i'; ...
>plot3d(i*j^2+100*normal(20,20),>zscale,scale=[1,1,1.5],angle=-40°,zoom=1.8
```




```
>Z=intrandom(5,100,6); v=zeros(5,6); ...
>loop 1 to 5; v[#]=getmultiplicities(1:6,Z[#]); end; ...
>columnsplot3d(v',scols=1:5,ccols=[1:5]):
```

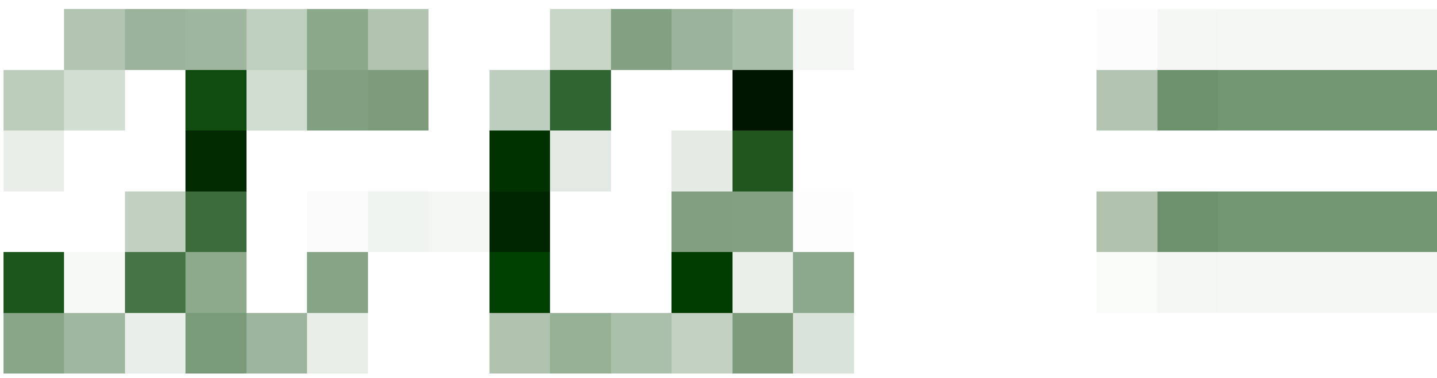


Contoh Soal:
Buatlah grafik berdasarkan data berikut.

$$x = -1 : 0.1 : 1; y = x'; z = x^2 + y;$$

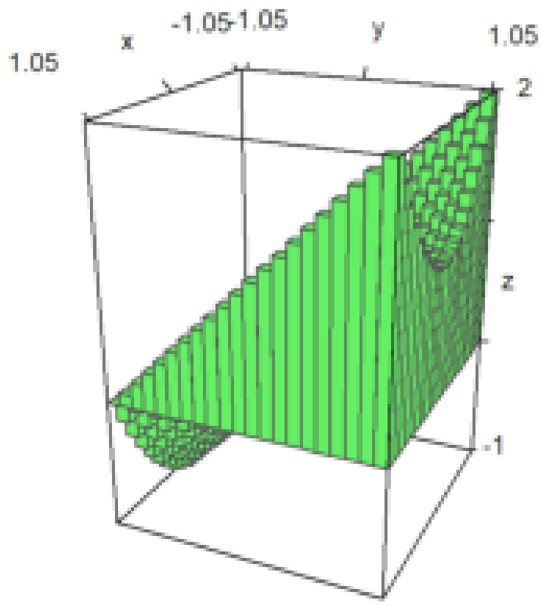
$$xa = (x|1.1) - 0.05; ya = (7_1.1) - 0.05$$

```
>x=-1:0.1:1; y=x'; z=x^2+y; ...
>xa=(x|1.1)-0.05; ya=(y_1.1)-0.05; ...
>plot3d(xa,ya,z,bar=true):
```



Permukaan Benda Putar

```
>plot2d("(x^2+y^2-1)^3-x^2*y^3",r=1.3, ...
>style="#",color=red,<outline, ...
>level=[-2;0],n=100):
```




```
>ekspresi &= (x^2+y^2-1)^3-x^2*y^3; $ekspresi
```

$$\left(y^2+x^2-1\right)^3-x^2\,y^3$$

Kami ingin memutar kurva hati di sekitar sumbu y. Inilah ungkapan yang mendefinisikan hati:

$$f(x,y)=(x^2+y^2-1)^3-x^2.y^3.$$

Selanjutnya kita menetepkan

$$x=r.cos(a),\quad y=r.sin(a).$$

```
>function fr(r,a) &= ekspresi with [x=r*cos(a),y=r*sin(a)] | trigreduce; $f
```

$$\left(r^2-1\right)^3+\frac{\left(\sin\left(5\,a\right)-\sin\left(3\,a\right)-2\,\sin a\right)\,r^5}{16}$$

Hal ini memungkinkan untuk mendefinisikan fungsi numerik, yang menyelesaikan r, jika a diberikan. Dengan fungsi tersebut kita dapat memplot jantung yang diputar sebagai permukaan parametrik.

```
>function map f(a) := bisect("fr",0,2;a); ...
>t=linspace(-pi/2,pi/2,100); r=f(t);  ...
>s=linspace(pi,2pi,100)'; ...
>plot3d(r*cos(t)*sin(s),r*cos(t)*cos(s),r*sin(t), ...
>>hue,<frame,color=red,zoom=4,amb=0,max=0.7,grid=12,height=50°):
```

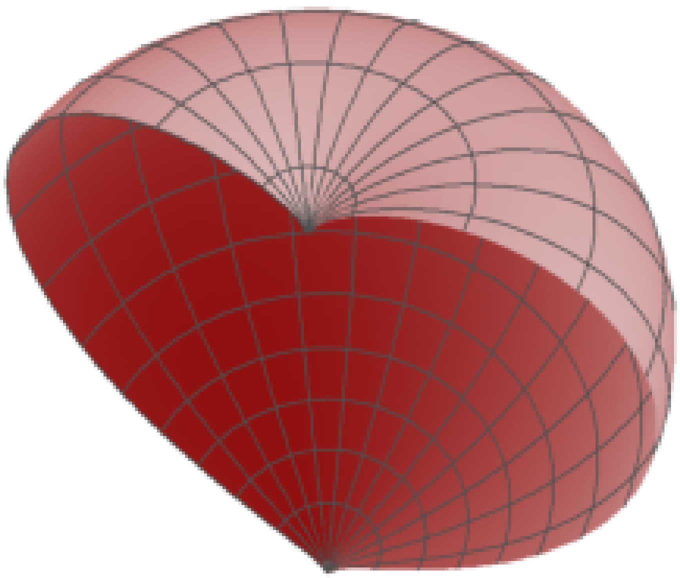
$$(r^2 - 1)^3$$

Berikut ini adalah plot 3D dari gambar di atas yang diputar mengelilingi sumbu z. Kami mendefinisikan fungsi yang mendeskripsikan objek.

```
>function f(x,y,z) ...
```

```
  r=x^2+y^2;
  return (r+z^2-1)^3-r*z^3;
   endfunction
```

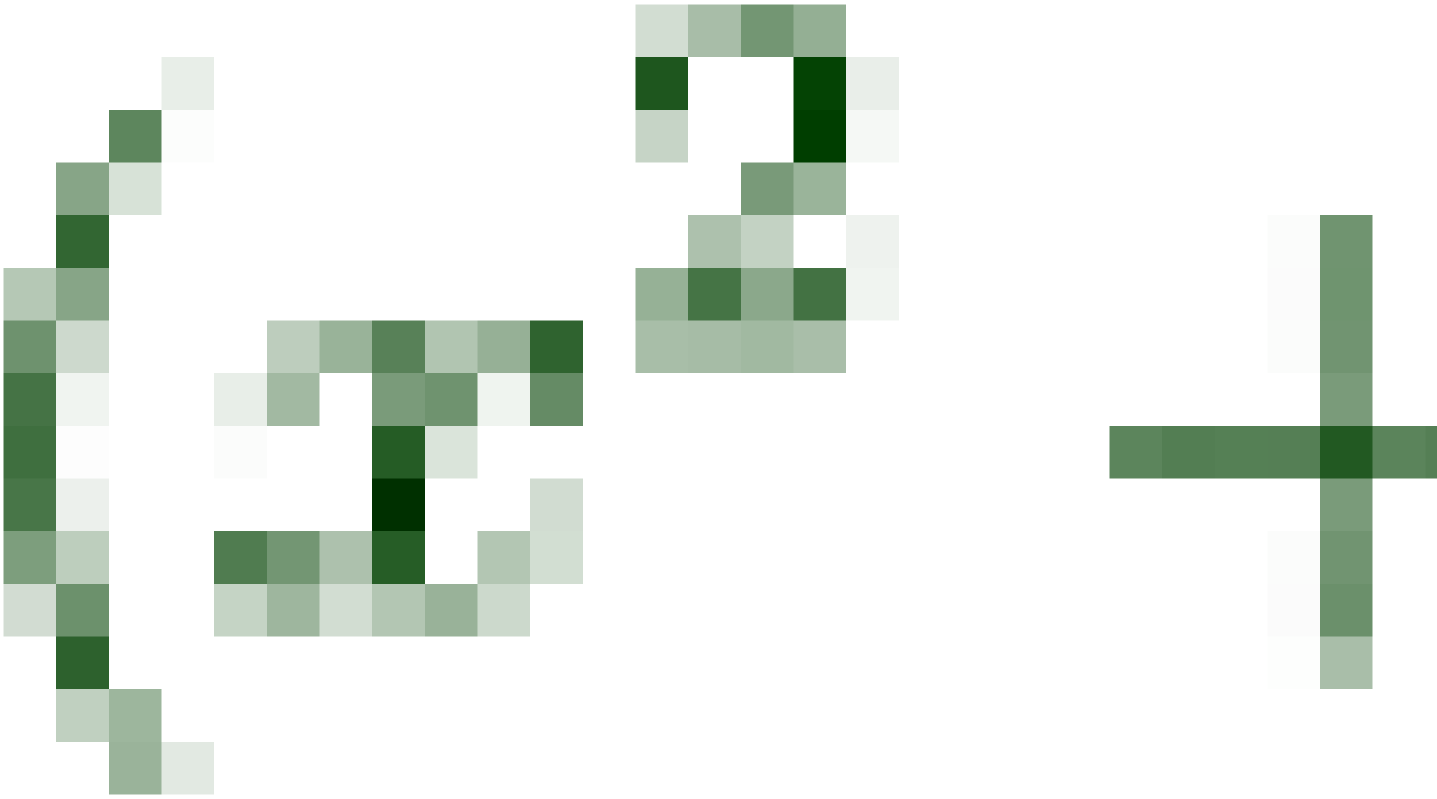
```
>plot3d("f(x,y,z)", ...
>xmin=0,xmax=1.2,ymin=-1.2,ymax=1.2,zmin=-1.2,zmax=1.4, ...
>implicit=1,angle=-30°,zoom=2.5,n=[10,100,60],>anaglyph):
```



Contoh Soal:
Gambarlah permukaan benda putar dari fungsi berikut ini.

$$(x^2 + y^2)^3 - x^2y^3$$

```
>plot2d("(x^2+y^2)^3-x^2*y^3",r=0.2, ...
>style="#",color=blue,<outline, ...
>level=[-2;0],n=100):
```



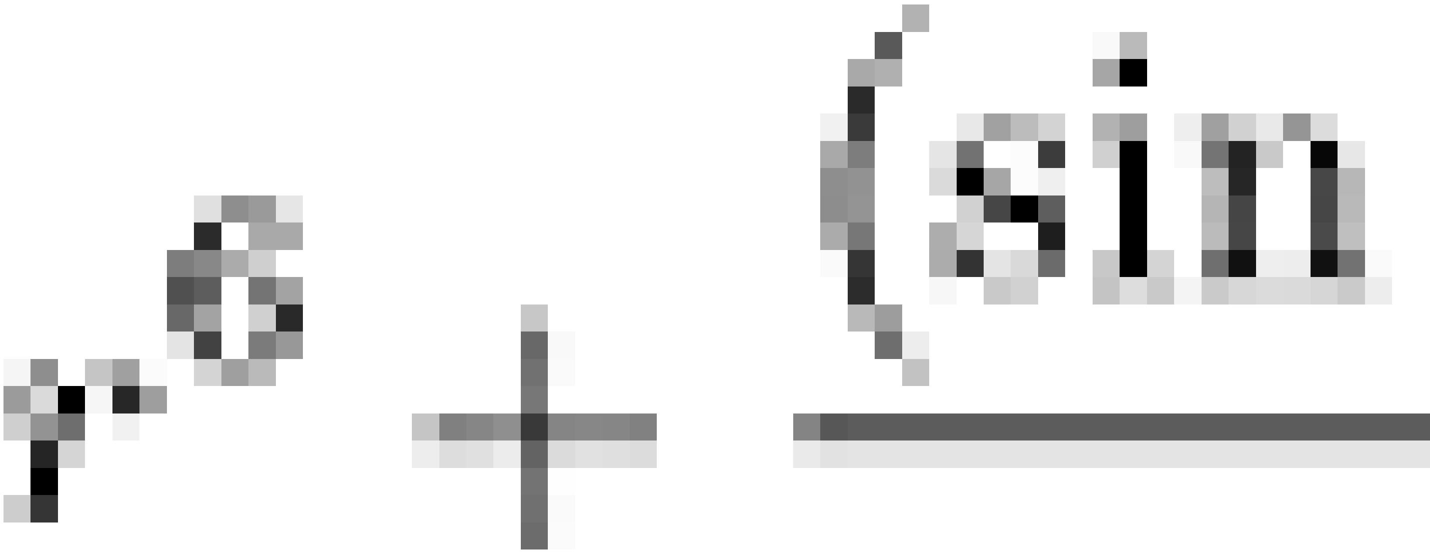
```
>ekspresi &= (x^2+y^2)^3-x^2*y^3; $ekspresi
```

$$\left(y^2+x^2\right)^3-x^2\,y^3$$

```
>function fr(r,a) &= ekspresi with [x=r*cos(a),y=r*sin(a)] | trigreduce; $f
```

$$r^6+\frac{\left(\sin \left(5\,a\right)-\sin \left(3\,a\right)-2\,\sin a\right)\,r^5}{16}$$

```
>function map f(a) := bisect("fr",0,2;a); ...
>t=linspace(-pi/2,pi/2,100); r=f(t);  ...
>s=linspace(pi,2pi,100)'; ...
>plot3d(r*cos(t)*sin(s),r*cos(t)*cos(s),r*sin(t), ...
>>hue,<frame,color=blue,zoom=4,amb=0,max=0.7,grid=12,height=50°):
```



Plot 3D Khusus

Fungsi plot3d bagus untuk dimiliki, tetapi tidak memenuhi semua kebutuhan. Selain rutinitas yang lebih mendasar, dimungkinkan untuk mendapatkan plot berbingkai dari objek apa pun yang Anda suka.

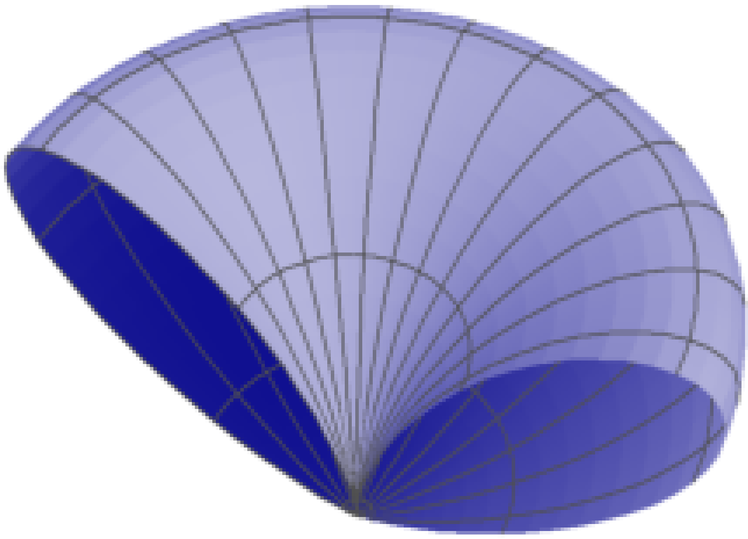
Meskipun Euler bukan program 3D, ia dapat menggabungkan beberapa objek dasar. Kami mencoba memvisualisasikan paraboloid dan garis singgungnya.

```
>function myplot ...
```

```
    y=-1:0.01:1; x=(-1:0.01:1)';
    plot3d(x,y,0.2*(x-0.1)/2,<scale,<frame,>hue, ..
      hues=0.5,>contour,color=orange);
    h=holding(1);
    plot3d(x,y,(x^2+y^2)/2,<scale,<frame,>contour,>hue);
    holding(h);
  endfunction
```

Sekarang framedplot() menyediakan frame, dan mengatur tampilan.

```
>framedplot("myplot",[-1,1,-1,1,0,1],height=0,angle=-30°, ...
>   center=[0,0,-0.7],zoom=3):
```

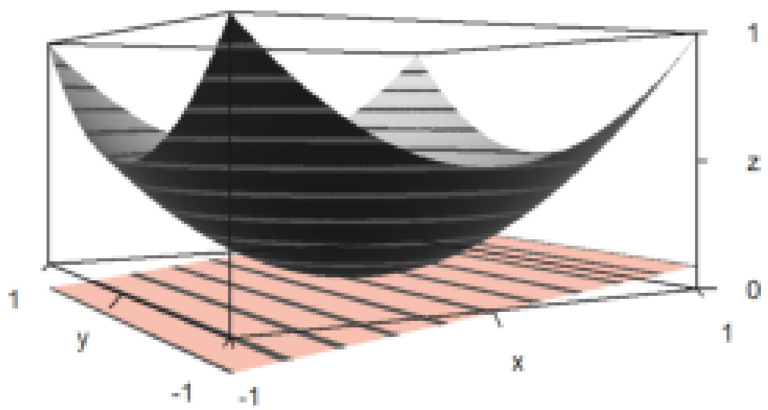


Dengan cara yang sama, Anda dapat memplot bidang kontur secara manual. Perhatikan bahwa plot3d() menyetel jendela ke fullwindow(), secara default, tetapi plotcontourplane() berasumsi demikian.

```
>x=-1:0.02:1.1; y=x'; z=x^2-y^4;
>function myplot (x,y,z) ...
```

```
    zoom(2);
    wi=fullwindow();
    plotcontourplane(x,y,z,level="auto",<scale);
    plot3d(x,y,z,>hue,<scale,>add,color=white,level="thin");
    window(wi);
    reset();
  endfunction
```

```
>myplot(x,y,z):
```



Contoh soal:
Buatlah plot bidang kontur dari fungsi berikut.

$$z = y^2 - x^2$$

```
>x=-1:0.02:1.1; y=x'; z=y^2-x^2;
>function myplot (x,y,z) ...
```

```
    zoom(2.5);
    wi=fullwindow();
    plotcontourplane(x,y,z,level="auto",<scale);
    plot3d(x,y,z,>hue,<scale,>add,color=green,level="thin");
    window(wi);
    reset();
  endfunction
```

```
>myplot(x,y,z):
```

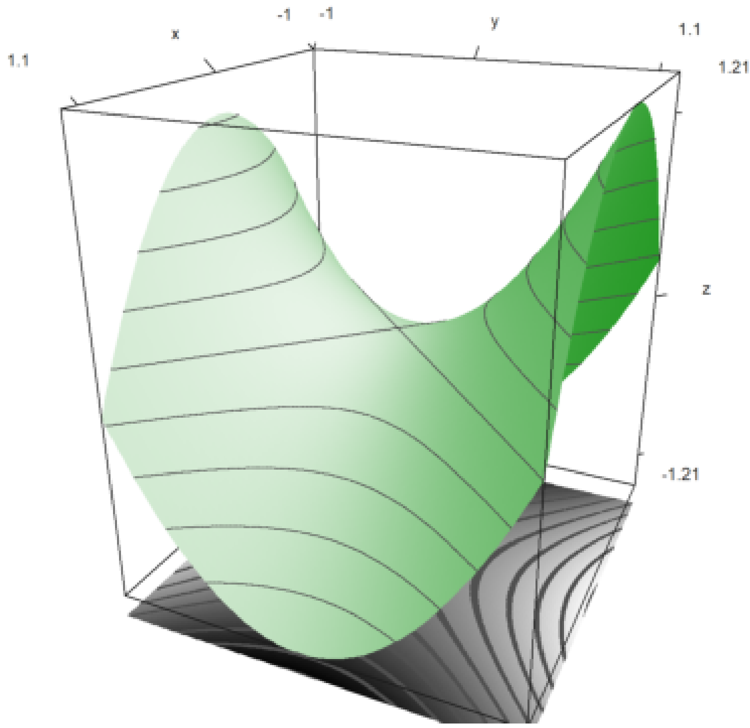
Animasi

Euler dapat menggunakan frame untuk melakukan pra-komputasi animasi.

Salah satu fungsi yang memanfaatkan teknik ini adalah memutar. Itu dapat mengubah sudut pandang dan menggambar ulang plot 3D. Fungsi ini memanggil addpage() untuk setiap plot baru. Akhirnya ia menganimasikan plotnya.

Silakan pelajari sumber rotasi untuk melihat lebih detail.

```
>function testplot () := plot3d("x^2+y^3"); ...
>rotate("testplot"); testplot():
```



Contoh Soal:
Gambarlah grafik dari fungsi berikut.

$$f(x, y) = -x^2 - y^2$$

```
>function testplot () := plot3d("exp(-x^2-y^2)"); ...
>rotate("testplot"); testplot():
```

Menggambar Povray

Dengan bantuan file Euler povray.e, Euler dapat menghasilkan file Povray. Hasilnya sangat bagus untuk dilihat.

Anda perlu menginstal Povray (32bit atau 64bit) dari http://www.povray.org/, dan meletakkan sub-direktori "bin" Povray ke jalur lingkungan, atau mengatur variabel "defaultpovray" dengan jalur lengkap yang mengarah ke "pvengine.exe".

Antarmuka Povray Euler menghasilkan file Povray di direktori home pengguna, dan memanggil Povray untuk menguraikan file-file ini. Nama file default adalah current.pov, dan direktori default adalah eulerhome(), biasanya c:\Users\Username\Euler. Povray menghasilkan file PNG, yang dapat dimuat oleh Euler ke dalam notebook. Untuk membersihkan file-file ini, gunakan povclear().

Fungsi pov3d memiliki semangat yang sama dengan plot3d. Ini dapat menghasilkan grafik fungsi f(x,y), atau permukaan dengan koordinat X,Y,Z dalam matriks, termasuk garis level opsional. Fungsi ini memulai raytracer secara otomatis, dan memuat adegan ke dalam notebook Euler.

Selain pov3d(), ada banyak fungsi yang menghasilkan objek Povray. Fungsi-fungsi ini mengembalikan string, yang berisi kode Povray untuk objek. Untuk menggunakan fungsi ini, mulai file Povray dengan povstart(). Kemudian gunakan writeln(...) untuk menulis objek ke file adegan. Terakhir, akhiri file dengan povend(). Secara default, raytracer akan dimulai, dan PNG akan dimasukkan ke dalam notebook Euler.

Fungsi objek memiliki parameter yang disebut "tampilan", yang memerlukan string dengan kode Povray untuk tekstur dan penyelesaian objek. Fungsi povlook() dapat digunakan untuk menghasilkan string ini. Ini memiliki parameter untuk warna, transparansi, Phong Shading dll.

Perhatikan bahwa alam semesta Povray memiliki sistem koordinat lain. Antarmuka ini menerjemahkan semua koordinat ke sistem Povray. Jadi Anda dapat terus berpikir dalam sistem koordinat Euler dengan z menunjuk vertikal ke atas, dan sumbu x,y,z di tangan kanan. Fungsi pov3d memiliki semangat yang sama dengan plot3d. Ini dapat menghasilkan grafik fungsi f(x,y), atau permukaan dengan koordinat X,Y,Z dalam matriks, termasuk garis level opsional. Fungsi ini memulai raytracer secara otomatis, dan memuat adegan ke dalam notebook Euler.
Anda perlu memuat file povray

```
>load povray;
```

Pastikan, direktori Povray bin ada di jalurnya. Jika tidak, edit variabel berikut sehingga berisi jalur ke povray yang dapat dieksekusi.

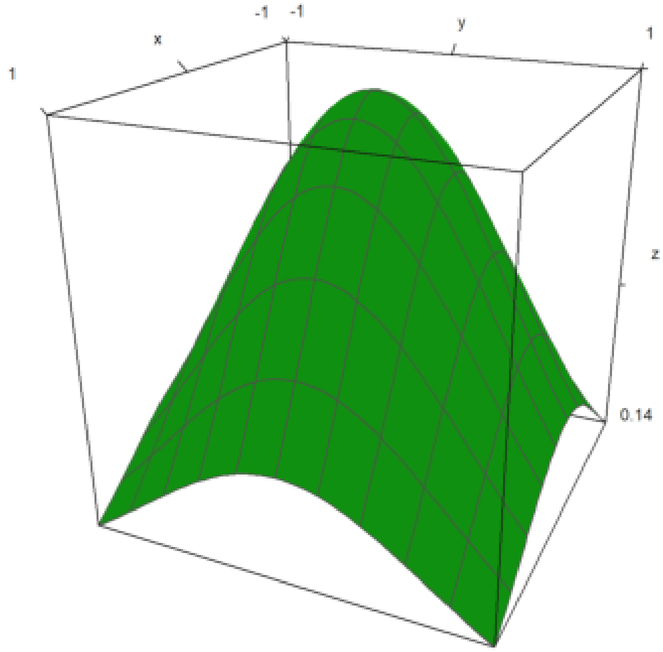
```
>defaultpovray="C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe"
```

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

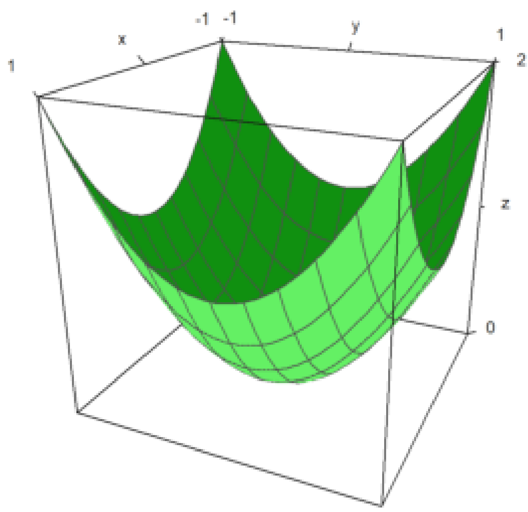
Untuk kesan pertama, kami memplot fungsi sederhana. Perintah berikut menghasilkan file povray di direktori pengguna Anda, dan menjalankan Povray untuk penelusuran sinar file ini.

ika Anda memulai perintah berikut, GUI Povray akan terbuka, menjalankan file, dan menutup secara otomatis. Karena alasan keamanan, Anda akan ditanya apakah Anda ingin mengizinkan file exe dijalankan. Anda dapat menekan batal untuk menghentikan pertanyaan lebih lanjut. Anda mungkin harus menekan OK di jendela Povray untuk mengonfirmasi dialog pengaktifan Povray.

```
>plot3d("x^2+y^2",zoom=2):
```

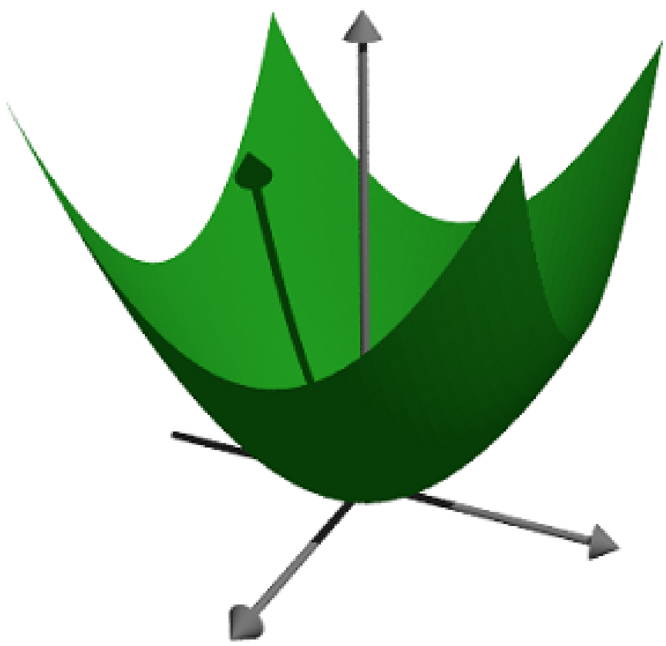



```
>pov3d("x^2+y^2",zoom=3);
```



Kita dapat membuat fungsinya transparan dan menambahkan penyelesaian lainnya. Kita juga dapat menambahkan garis level ke plot fungsi.

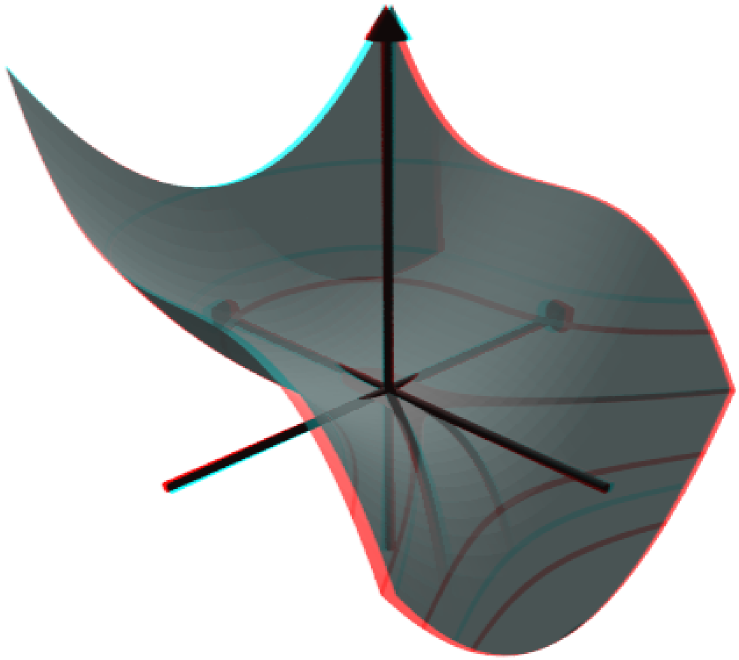
```
>pov3d("x^2+y^3",axiscolor=red,angle=-45°,>anaglyph, ...
>  look=povlook(cyan,0.2),level=-1:0.5:1,zoom=3.8);
```



Terkadang perlu untuk mencegah penskalaan fungsi, dan menskalakan fungsi secara manual.

Kita memplot himpunan titik pada bidang kompleks, dimana hasil kali jarak ke 1 dan -1 sama dengan 1.

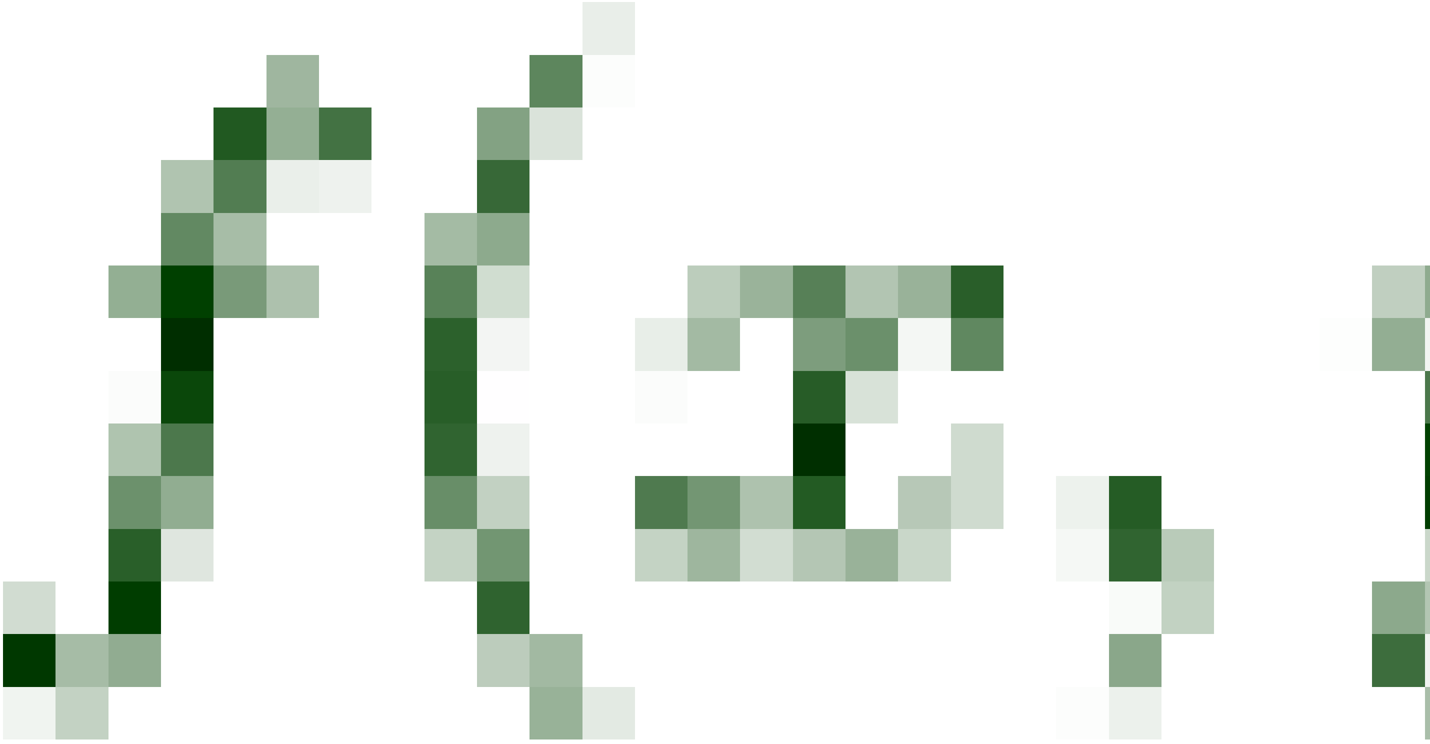
```
>pov3d("((x-1)^2+y^2)*((x+1)^2+y^2)/40",r=2, ...
>  angle=-120°,level=1/40,dlevel=0.005,light=[-1,1,1],height=10°,n=50, ...
>  <fscale,zoom=3.8);
```



Contoh Soal:
Gambarlah povray dari fungsi berikut.

$$f(x, y) = 3 - x^2 - y^2$$

```
>pov3d("3-x^2-y^2",zoom=3);
```



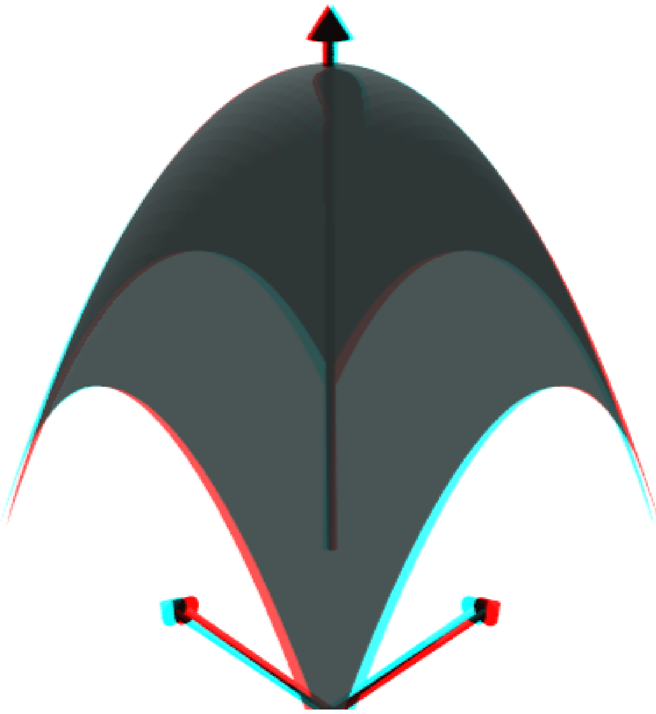
```
>pov3d("3-x^2-y^2",axiscolor=red,angle=-45°,>anaglyph, ...
>  look=povlook(cyan,0.2),level=-1:0.5:1,zoom=3.8);
```

Merencanakan dengan Koordinat

Daripada menggunakan fungsi, kita bisa memplotnya dengan koordinat. Seperti di plot3d, kita memerlukan tiga matriks untuk mendefinisikan objek.

Dalam contoh ini kita memutar suatu fungsi di sekitar sumbu z.

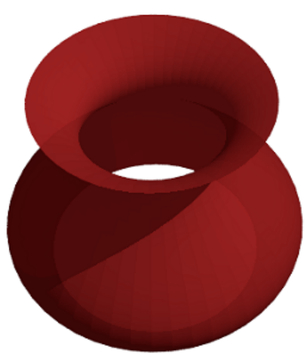
```
>function f(x) := x^3-x+1; ...
>x=-1:0.01:1; t=linspace(0,2pi,50)'; ...
>Z=x; X=cos(t)*f(x); Y=sin(t)*f(x); ...
>pov3d(X,Y,Z,angle=40°,look=povlook(red,0.1),height=50°,axis=0,zoom=4,light
```



Pada contoh berikut, kita memplot gelombang teredam. Kami menghasilkan gelombang dengan bahasa matriks Euler.

Kami juga menunjukkan, bagaimana objek tambahan dapat ditambahkan ke adegan pov3d. Untuk pembuatan objek, lihat contoh berikut. Perhatikan bahwa plot3d menskalakan plot, sehingga cocok dengan kubus satuan.

```
>r=linspace(0,1,80); phi=linspace(0,2pi,80)'; ...
>x=r*cos(phi); y=r*sin(phi); z=exp(-5*r)*cos(8*pi*r)/3;  ...
>pov3d(x,y,z,zoom=6,axis=0,height=30°,add=povsphere([0.5,0,0.25],0.15,povlo
>  w=500,h=300);
```



Dengan metode peneduh canggih Povray, sangat sedikit titik yang dapat menghasilkan permukaan yang sangat halus. Hanya pada batas-batas dan dalam bayangan, triknya mungkin terlihat jelas.

Untuk ini, kita perlu menjumlahkan vektor normal di setiap titik matriks.

```
>Z &= x^2*y^3
```

```
                                 2  3
                                x  y
```

Persamaan permukaannya adalah [x,y,Z]. Kami menghitung dua turunan dari x dan y dan mengambil perkalian silangnya sebagai normal.

```
>dx &= diff([x,y,Z],x); dy &= diff([x,y,Z],y);
```

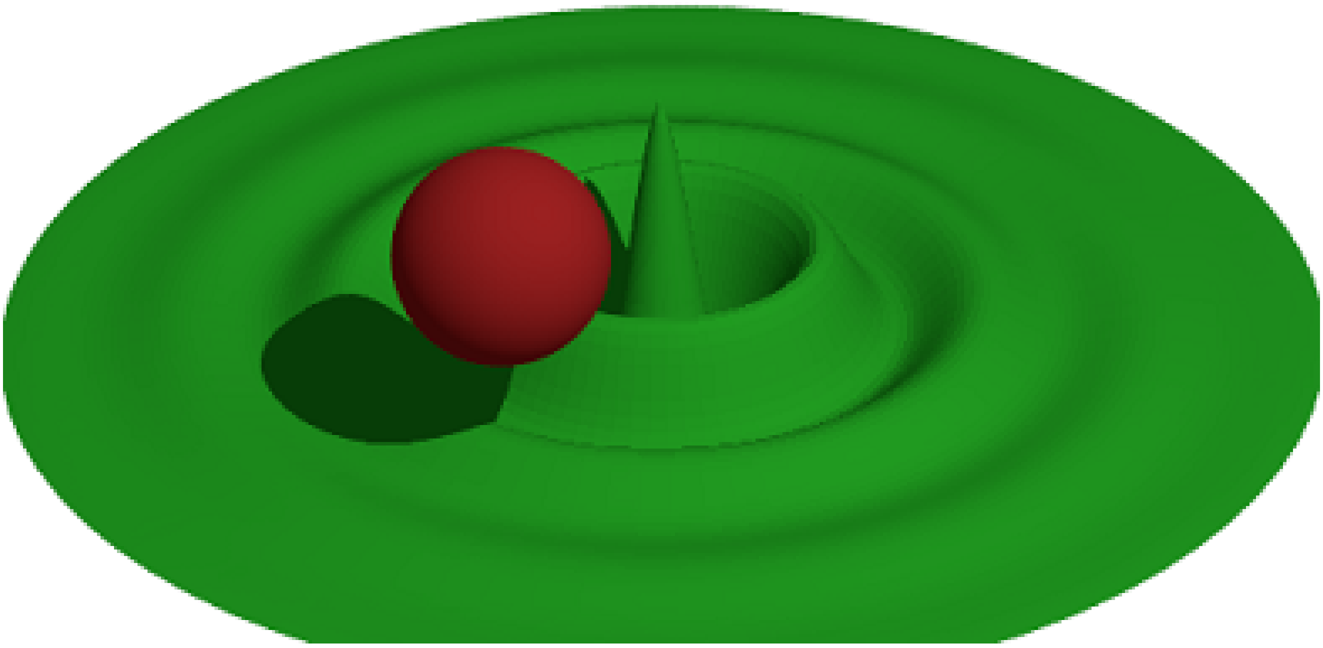
Kami mendefinisikan normal sebagai produk silang dari turunan ini, dan mendefinisikan fungsi koordinat

```
>N &= crossproduct(dx,dy); NX &= N[1]; NY &= N[2]; NZ &= N[3]; N,
```

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

Kami hanya menggunakan 25 poin.

```
>x=-1:0.5:1; y=x';
>pov3d(x,y,Z(x,y),angle=10°, ...
>  xv=NX(x,y),yv=NY(x,y),zv=NZ(x,y),<shadow);
```



Berikut adalah simpul Trefoil yang dibuat oleh A. Busser dalam Povray. Ada versi yang diperbarui dari ini dalam contoh-contoh.

Lihat: Examples\Trefoil Knot | Trefoil Knot

Untuk tampilan yang bagus dengan jumlah titik yang tidak terlalu banyak, kami menambahkan vektor normal di sini. Kami menggunakan Maxima untuk menghitung normalnya. Pertama, tiga fungsi koordinat sebagai ungkapan simbolis.

```
>X &= ((4+sin(3*y))+cos(x))*cos(2*y); ...
>Y &= ((4+sin(3*y))+cos(x))*sin(2*y); ...
>Z &= sin(x)+2*cos(3*y);
```

Kemudian dua vektor turunan terhadap x dan y.

```
>dx &= diff([X,Y,Z],x); dy &= diff([X,Y,Z],y);
```

Sekarang normalnya, yang merupakan hasil perkalian silang dari kedua turunan tersebut.

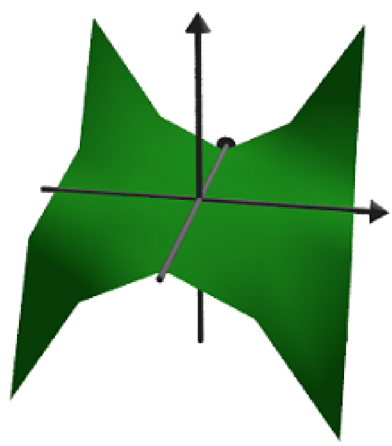
```
>dn &= crossproduct(dx,dy);
```

Kami sekarang mengevaluasi semua ini secara numerik.

```
>x:=linspace(-%pi,%pi,40); y:=linspace(-%pi,%pi,100)';
```

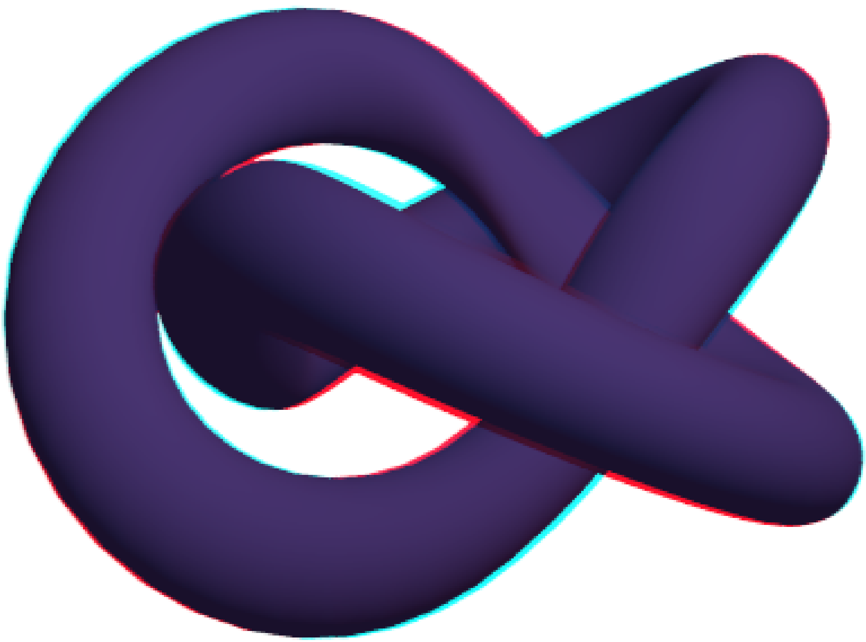
Vektor normal adalah hasil evaluasi dari ekspresi simbolis dn[i] untuk i=1,2,3. Syntax untuk ini adalah &"ekspresi"(parameter). Ini merupakan alternatif dari metode pada contoh sebelumnya, di mana kita mendefinisikan ekspresi simbolis NX, NY, NZ terlebih dahulu.

```
>pov3d(X(x,y),Y(x,y),Z(x,y),>anaglyph,axis=0,zoom=5,w=450,h=350, ...
>  <shadow,look=povlook(blue), ...
>  xv=&"dn[1]"(x,y), yv=&"dn[2]"(x,y), zv=&"dn[3]"(x,y));
```



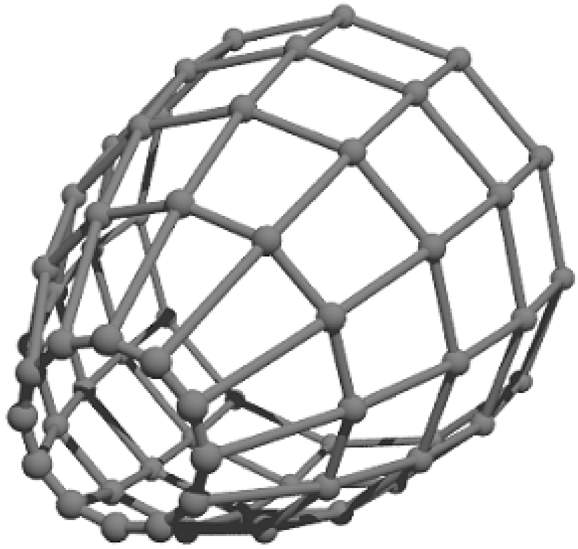
Kita juga dapat membuat grid dalam 3D.

```
>povstart(zoom=4); ...
>x=-1:0.5:1; r=1-(x+1)^2/6; ...
>t=(0°:30°:360°)'; y=r*cos(t); z=r*sin(t); ...
>writeln(povgrid(x,y,z,d=0.02,dballs=0.05)); ...
>povend();
```



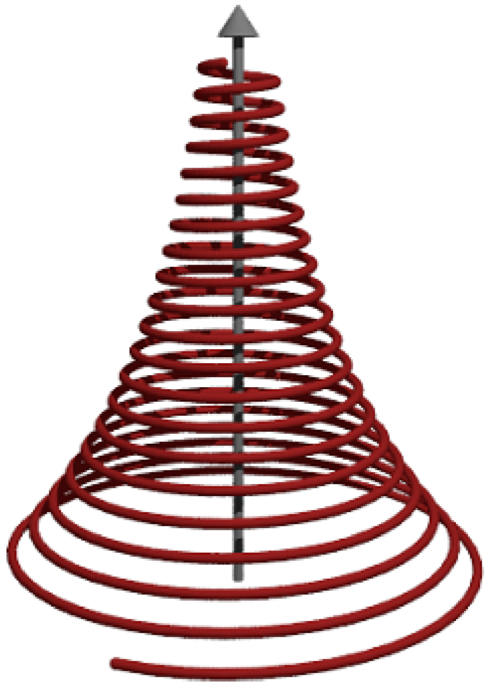
Dengan povgrid(), kurva-kurva menjadi mungkin.

```
>povstart(center=[0,0,1],zoom=3.6); ...
>t=linspace(0,2,1000); r=exp(-t); ...
>x=cos(2*pi*10*t)*r; y=sin(2*pi*10*t)*r; z=t; ...
>writeln(povgrid(x,y,z,povlook(red))); ...
>writeAxis(0,2,axis=3); ...
>povend();
```



Contoh lain membuat grid 3d

```
>povstart(zoom=2); ...
>x=-1:0.5:1; r=3-(x+2)^1/2; ...
>t=(0°:30°:360°)'; y=r*cos(t); z=r*sin(t); ...
>writeln(povgrid(x,y,z,d=0.02,dballs=0.05)); ...
>povend();
```



Objek Povray

Di atas, kami menggunakan pov3d untuk memplot permukaan. Antarmuka povray di Euler juga dapat menghasilkan objek Povray. Objek ini disimpan sebagai string di Euler, dan perlu ditulis ke file Povray.

Kami memulai output dengan povstart().

```
>povstart(zoom=4);
```

Pertama kita mendefinisikan tiga silinder, dan menyimpannya dalam string di Euler.

Fungsi povx() dll. hanya mengembalikan vektor [1,0,0], yang dapat digunakan sebagai gantinya.

```
>c1=povcylinder(-povx,povx,1,povlook(red)); ...
>c2=povcylinder(-povy,povy,1,povlook(yellow)); ...
>c3=povcylinder(-povz,povz,1,povlook(blue)); ...
```

String tersebut berisi kode Povray, yang tidak perlu kita pahami pada saat itu.

Fungsi povx() dll. hanya mengembalikan vektor [1,0,0], yang dapat digunakan sebagai gantinya.

```
>c2
```

```
 cylinder { <0,0,-1>, <0,0,1>, 1
  texture { pigment { color rgb <0.941176,0.941176,0.392157> }  }
  finish { ambient 0.2 }
  }
```

As you see, we added texture to the objects in three different colors.

Hal ini dilakukan oleh povlook(), yang mengembalikan string dengan kode Povray yang relevan. Kita dapat menggunakan warna default Euler, atau menentukan warna kita sendiri. Kita juga dapat menambahkan transparansi, atau mengubah cahaya sekitar.

```
>povlook(rgb(0.1,0.2,0.3),0.1,0.5)
```

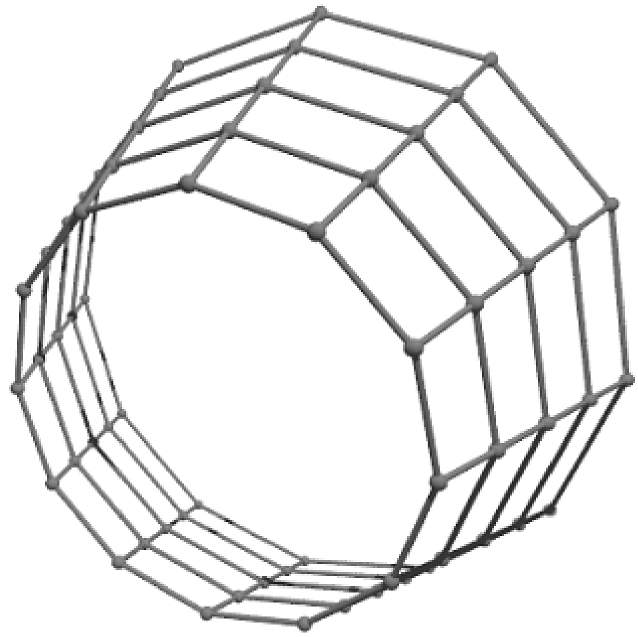
```
  texture { pigment { color rgbf <0.101961,0.2,0.301961,0.1> }  }
  finish { ambient 0.5 }
```

Sekarang kita mendefinisikan objek persimpangan, dan menulis hasilnya ke file.
i dilakukan oleh povlook(), yang mengembalikan string dengan kode Povray yang relevan. Kita dapat menggunakan warna default Euler, atau menentukan warna kita sendiri. Kita juga dapat menambahkan transparansi, atau mengubah cahaya sekitar.

```
>writeln(povintersection([c1,c2,c3]));
```

Persimpangan tiga silinder sulit untuk divisualisasikan jika Anda belum pernah melihatnya sebelumnya.

```
>povend;
```



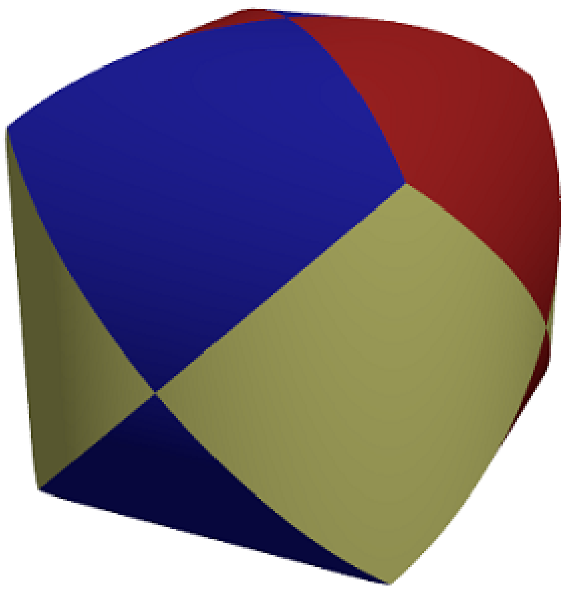
Fungsi berikut menghasilkan fraktal secara rekursif.

Fungsi pertama menunjukkan bagaimana Euler menangani objek Povray sederhana. Fungsi povbox() mengembalikan string, yang berisi koordinat kotak, tekstur, dan hasil akhir.

```
>function onebox(x,y,z,d) := povbox([x,y,z],[x+d,y+d,z+d],povlook());
>function fractal (x,y,z,h,n) ...
```

```
  if n==1 then writeln(onebox(x,y,z,h));
  else
    h=h/3;
    fractal(x,y,z,h,n-1);
    fractal(x+2*h,y,z,h,n-1);
    fractal(x,y+2*h,z,h,n-1);
    fractal(x,y,z+2*h,h,n-1);
    fractal(x+2*h,y+2*h,z,h,n-1);
    fractal(x+2*h,y,z+2*h,h,n-1);
    fractal(x,y+2*h,z+2*h,h,n-1);
    fractal(x+2*h,y+2*h,z+2*h,h,n-1);
    fractal(x+h,y+h,z+h,h,n-1);
  endif;
endfunction
```

```
>povstart(fade=10,<shadow);
>fractal(-1,-1,-1,2,4);
>povend();
```



Perbedaan memungkinkan pemisahan satu objek dari objek lainnya. Seperti persimpangan, ada bagian dari objek CSG di Povray.

```
>povstart(light=[5,-5,5],fade=10);
```

Untuk demonstrasi ini, kita akan mendefinisikan sebuah objek di Povray, alih-alih menggunakan sebuah string di Euler. Definisi akan langsung dituliskan ke file.

Koordinat kotak -1 berarti [-1,-1,-1].

```
>povdefine("mycube",povbox(-1,1));
```

Kita dapat menggunakan objek ini dalam povobject(), yang mengembalikan sebuah string seperti biasa.

```
>c1=povobject("mycube",povlook(red));
```

Kami menghasilkan kubus kedua, dan memutar serta menskalakannya sedikit.

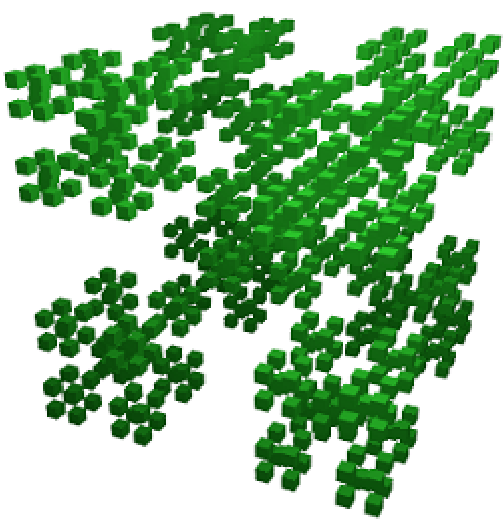
```
>c2=povobject("mycube",povlook(yellow),translate=[1,1,1], ...
>  rotate=xrotate(10°)+yrotate(10°), scale=1.2);
```

Kemudian kita ambil selisih dari kedua objek tersebut.

```
>writeln(povdifference(c1,c2));
```

Sekarang tambahkan tiga sumbu.

```
>writeAxis(-1.2,1.2,axis=1); ...
>writeAxis(-1.2,1.2,axis=2); ...
>writeAxis(-1.2,1.2,axis=4); ...
>povend();
```



Fungsi Implisit

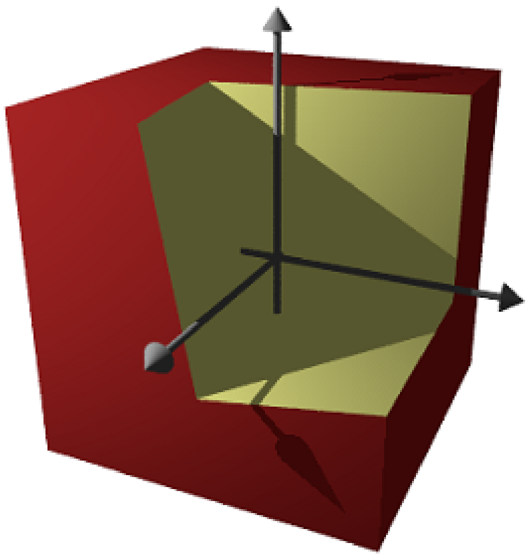
Povray dapat memplot himpunan di mana f(x,y,z)=0, seperti parameter implisit di plot3d. Namun hasilnya terlihat jauh lebih baik.

Sintaks untuk fungsinya sedikit berbeda. Anda tidak dapat menggunakan keluaran ekspresi Maxima atau Euler.

```
>povstart(angle=70°,height=50°,zoom=4);
```

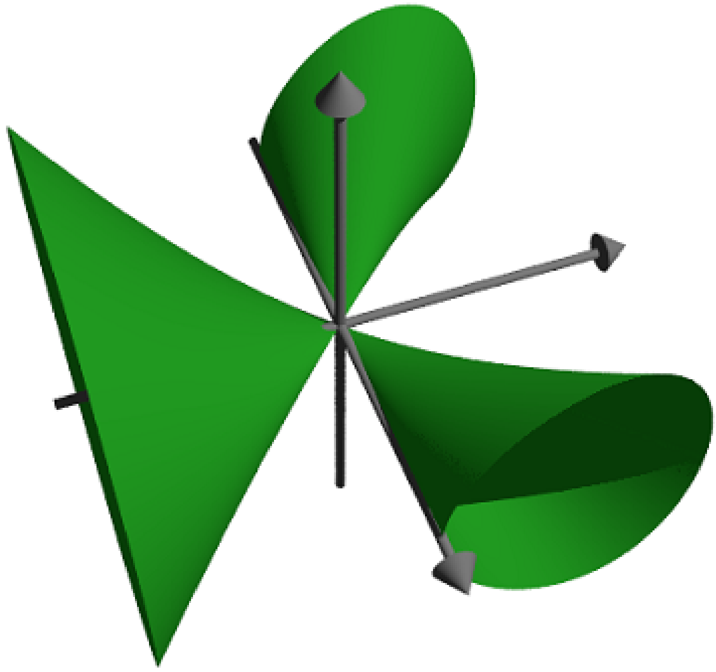
Buatlah permukaan implisit. Perhatikan sintaks yang berbeda pada ekspresi ini.

```
>writeln(povsurface("pow(x,2)*y-pow(y,3)-pow(z,2)",povlook(green))); ...
>writeAxes(); ...
>povend();
```



Contoh tambahan:
Buatlah permukaan implisit.

```
>povstart(angle=50°,height=50°,zoom=4);
>writeln(povsurface("pow(x,1)*y-pow(y,2)-pow(z,1)",povlook(white))); ...
>writeAxes(); ...
>povend();
```



Objek Jaring

Dalam contoh ini, kami menunjukkan cara membuat objek mesh, dan menggambarnya dengan informasi tambahan.

Kita ingin memaksimalkan xy pada kondisi x+y=1 dan mendemonstrasikan sentuhan tangensial garis datar.

```
>povstart(angle=-10°,center=[0.5,0.5,0.5],zoom=7);
```

Kita tidak dapat menyimpan objek dalam sebuah string seperti sebelumnya, karena ukurannya terlalu besar. Jadi kita mendefinisikan objek dalam file Povray menggunakan declare. Fungsi povtriangle() melakukan hal ini secara otomatis. Fungsi ini dapat menerima vektor normal seperti halnya pov3d().

Berikut ini mendefinisikan objek mesh, dan menuliskannya langsung ke dalam file.

```
>x=0:0.02:1; y=x'; z=x*y; vx=-y; vy=-x; vz=1;
>mesh=povtriangles(x,y,z,"",vx,vy,vz);
```

Sekarang kita tentukan dua cakram, yang akan berpotongan dengan permukaan.

```
>cl=povdisc([0.5,0.5,0],[1,1,0],2); ...
>ll=povdisc([0,0,1/4],[0,0,1],2);
```

Tuliskan permukaan dikurangi kedua cakram.

```
>writeln(povdifference(mesh,povunion([cl,ll]),povlook(green)));
```

Tuliskan kedua perpotongan tersebut.

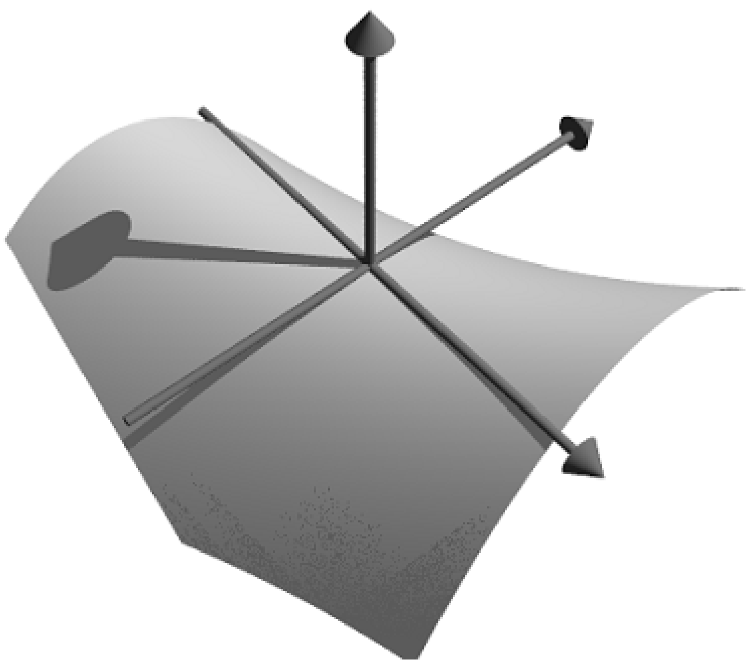
```
>writeln(povintersection([mesh,cl],povlook(red))); ...
>writeln(povintersection([mesh,ll],povlook(gray)));
```

Tulislah satu titik secara maksimal.

```
>writeln(povpoint([1/2,1/2,1/4],povlook(gray),size=2*defaultpointsize));
```

Tambahkan sumbu dan selesaikan.

```
>writeAxes(0,1,0,1,0,1,d=0.015); ...
>povend();
```

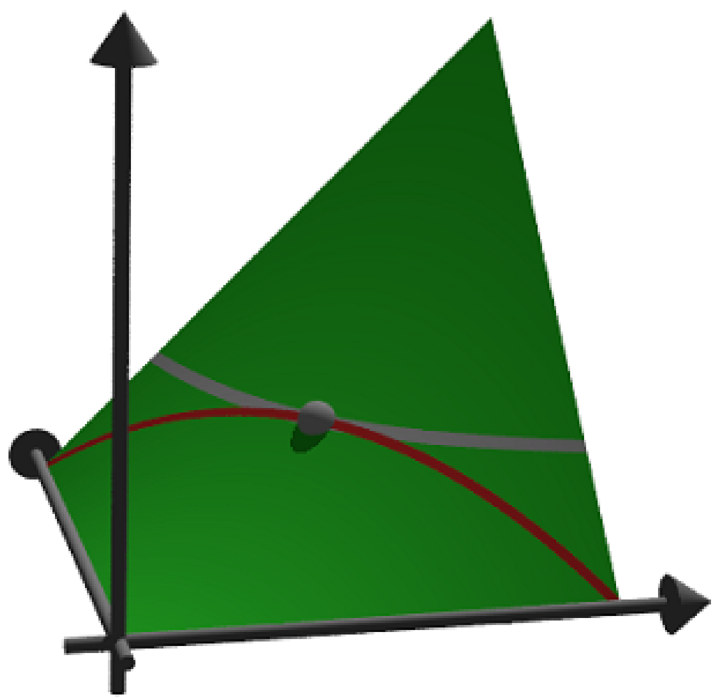


Untuk menghasilkan anaglyph untuk kacamata merah/cyan, Povray harus dijalankan dua kali dari posisi kamera berbeda. Ini menghasilkan dua file Povray dan dua file PNG, yang dimuat dengan fungsi loadanaglyph().

Tentu saja, Anda memerlukan kacamata berwarna merah/cyan untuk melihat contoh berikut dengan benar.

Fungsi pov3d() memiliki saklar sederhana untuk menghasilkan anaglyph.

```
>pov3d("-exp(-x^2-y^2)/2",r=2,height=45°,>anaglyph, ...
>  center=[0,0,0.5],zoom=3.5);
```



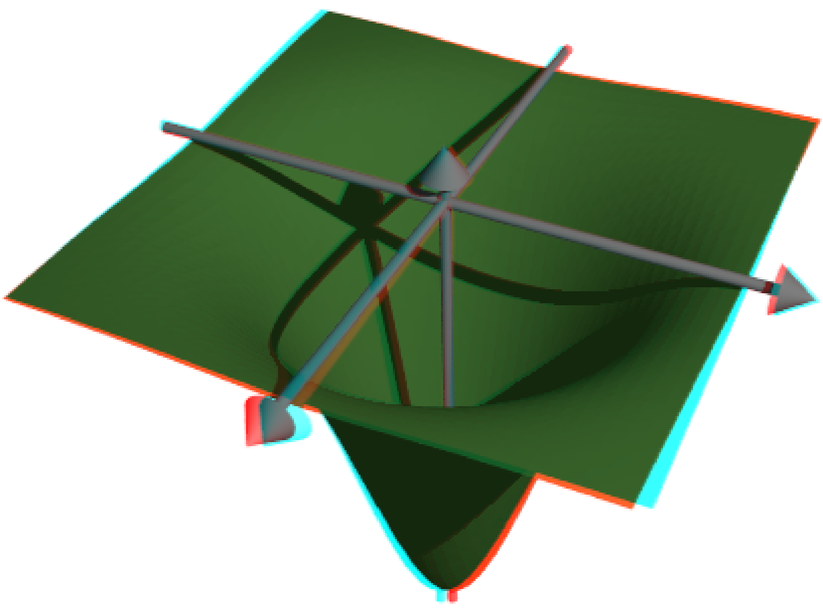
Jika Anda membuat scene dengan objek, Anda harus menempatkan pembuatan scene ke dalam suatu fungsi, dan menjalankannya dua kali dengan nilai yang berbeda untuk parameter anaglyph.

```
>function myscene ...
```

```
    s=povsphere(povc,1);
    cl=povcylinder(-povz,povz,0.5);
    clx=povobject(cl,rotate=xrotate(90°));
    cly=povobject(cl,rotate=yrotate(90°));
    c=povbox([-1,-1,0],1);
    un=povunion([cl,clx,cly,c]);
    obj=povdifference(s,un,povlook(red));
    writeln(obj);
    writeAxes();
  endfunction
```

Fungsi povanaglyph() melakukan semua ini. Parameter-parameternya seperti pada povstart() dan povend() yang digabungkan.

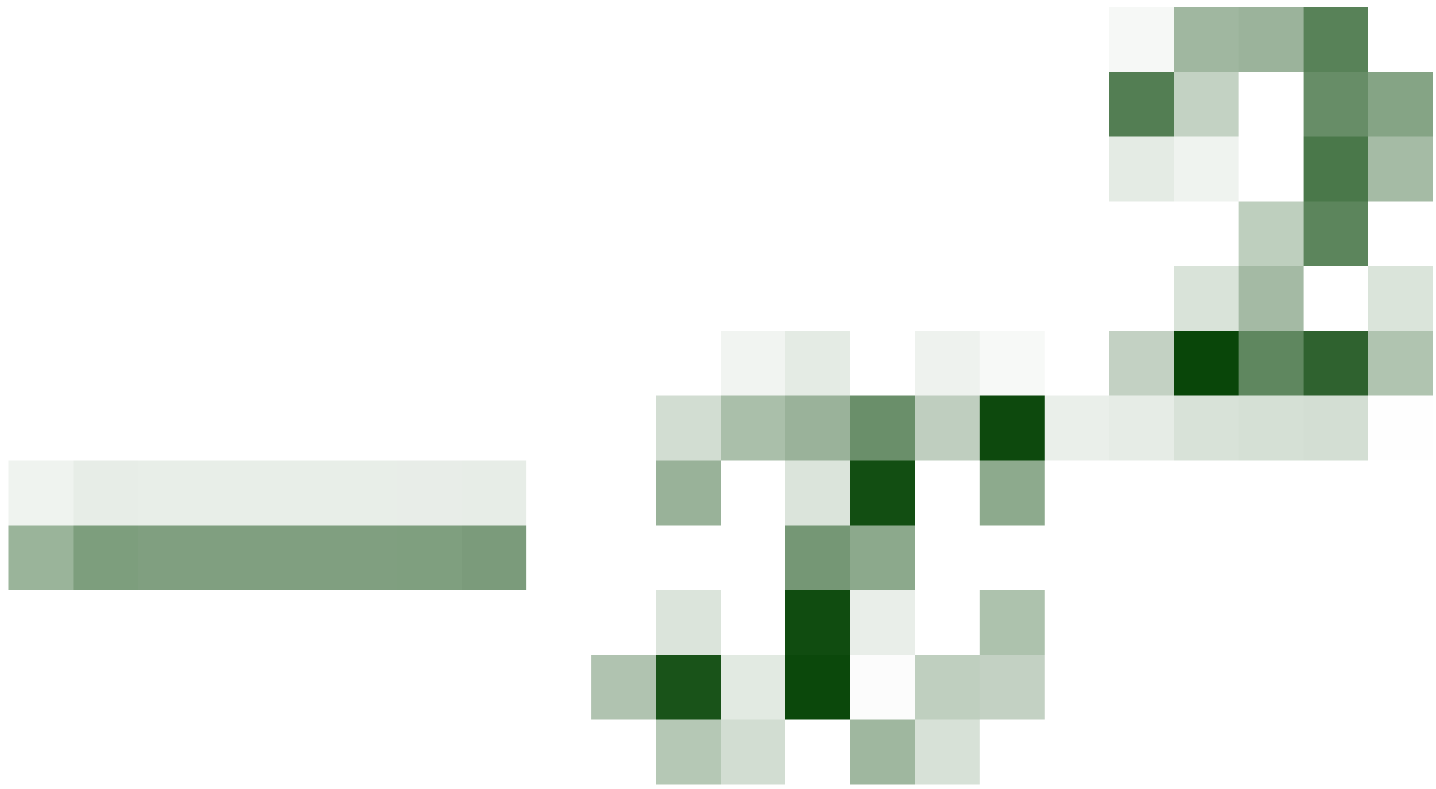
```
>povanaglyph("myscene",zoom=4.5);
```



Contoh Soal:
Buatlah anaglyph dari fungsi berikut.

$$-x^2 + y^2$$

```
>pov3d("-exp(-x^2+y^2)/3",r=2,height=45°,>anaglyph, ...
>center=[0,0,0.5],zoom=3.5);
```



Mendefinisikan Objek sendiri

Antarmuka povray Euler berisi banyak objek. Namun Anda tidak dibatasi pada hal ini. Anda dapat membuat objek sendiri, yang menggabungkan objek lain, atau merupakan objek yang benar-benar baru.

Kami mendemonstrasikan torus. Perintah Povray untuk ini adalah "torus". Jadi kami mengembalikan string dengan perintah ini dan parameternya. Perhatikan bahwa torus selalu berpusat pada titik asal.

```
>function povdonat (r1,r2,look="") ...
```

```
    return "torus {"+r1+","+r2+look+"}";
  endfunction
```

Inilah torus pertama kami.

```
>t1=povdonat(0.8,0.2)
```

```
torus {0.8,0.2}
```

Mari kita gunakan objek ini untuk membuat torus kedua, ditranslasikan dan diputar.

```
>t2=povobject(t1,rotate=xrotate(90°),translate=[0.8,0,0])
```

```
object { torus {0.8,0.2}
 rotate 90 *x
 translate <0.8,0,0>
 }
```

Sekarang, kita tempatkan semua benda ini ke dalam suatu pemandangan. Untuk tampilannya, kami menggunakan Phong Shading.

```
>povstart(center=[0.4,0,0],angle=0°,zoom=3.8,aspect=1.5); ...
>writeln(povobject(t1,povlook(green,phong=1))); ...
>writeln(povobject(t2,povlook(green,phong=1))); ...
```

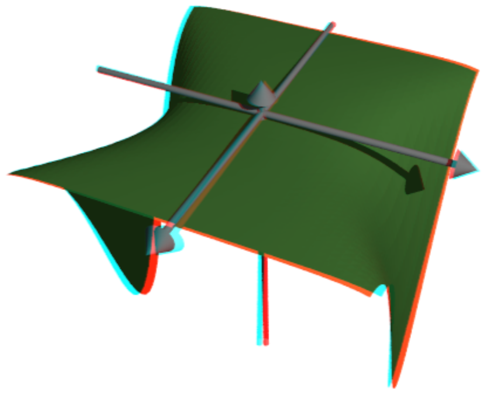
```
>povend();
```

memanggil program Povray. Namun, jika terjadi kesalahan, program ini tidak menampilkan kesalahan. Oleh karena itu, Anda harus menggunakan

```
>povend(<exit);
```

jika ada yang tidak berhasil. Ini akan membiarkan jendela Povray terbuka.

```
>povend(h=320,w=480);
```



Berikut adalah contoh yang lebih rumit. Kami menyelesaikan

$$Ax \leq b, \quad x \geq 0, \quad c.x \to \text{Max.}$$

dan menunjukkan titik-titik yang layak dan optimal dalam plot 3D.

```
>A=[10,8,4;5,6,8;6,3,2;9,5,6];
>b=[10,10,10,10]';
>c=[1,1,1];
```

Pertama, mari kita periksa, apakah contoh ini memiliki solusi atau tidak..

```
>x=simplex(A,b,c,>max,>check)'
```

```
[0,   1,   0.5]
```

Ya, benar.

Selanjutnya kita mendefinisikan dua objek. Yang pertama adalah pesawat

$$a \cdot x \leq b$$

```
>function oneplane (a,b,look="") ...
```

```
    return povplane(a,b,look)
  endfunction
```

Kemudian kita tentukan perpotongan semua setengah ruang dan kubus.

```
>function adm (A, b, r, look="") ...
```

```
    ol=[];
    loop 1 to rows(A); ol=ol|oneplane(A[#],b[#]); end;
    ol=ol|povbox([0,0,0],[r,r,r]);
    return povintersection(ol,look);
  endfunction
```

Sekarang, kita bisa merencanakan adegan tersebut.

```
>povstart(angle=120°,center=[0.5,0.5,0.5],zoom=3.5); ...
>writeln(adm(A,b,2,povlook(green,0.4))); ...
>writeAxes(0,1.3,0,1.6,0,1.5); ...
```

Berikut ini adalah lingkaran di sekeliling optimal.

```
>writeln(povintersection([povsphere(x,0.5),povplane(c,c.x')], ...
>  povlook(red,0.9)));
```

Dan kesalahan pada arah yang optimal.

```
>writeln(povarrow(x,c*0.5,povlook(red)));
```

Kami menambahkan teks ke layar. Teks hanyalah sebuah objek 3D. Kita perlu menempatkan dan memutarnya sesuai dengan pandangan kita.

```
>writeln(povtext("Linear Problem",[0,0.2,1.3],size=0.05,rotate=5°)); ...
>povend();
```

Contoh Lainnya

Anda dapat menemukan beberapa contoh Povray di Euler di file berikut.

ee: Examples/Dandelin Spheres
See: Examples/Donat Math
See: Examples/Trefoil Knot
See: Examples/Optimization by Affine Scaling

BAB 5

KALKULUS DENGAN EMT

Kalkulus dengan EMT

Materi Kalkulus mencakup di antaranya:

- Fungsi (fungsi aljabar, trigonometri, eksponensial, logaritma, komposisi fungsi)
- Limit Fungsi,
- Turunan Fungsi,
- Integral Tak Tentu,
- Integral Tentu dan Aplikasinya,
- Barisan dan Deret (kekonvergenan barisan dan deret).

EMT (bersama Maxima) dapat digunakan untuk melakukan semua perhitungan di dalam kalkulus, baik secara numerik maupun analitik (eksak).

Mendefinisikan Fungsi

Terdapat beberapa cara mendefinisikan fungsi pada EMT, yakni:

- Menggunakan format nama_fungsi := rumus fungsi (untuk fungsi numerik),
- Menggunakan format nama_fungsi &= rumus fungsi (untuk fungsi simbolik, namun dapat dihitung secara numerik),
- Menggunakan format nama_fungsi &&= rumus fungsi (untuk fungsi simbolik murni, tidak dapat dihitung langsung),
- Fungsi sebagai program EMT.

Setiap format harus diawali dengan perintah function (bukan sebagai ekspresi).

Berikut adalah adalah beberapa contoh cara mendefinisikan fungsi:

$$f(x) = 2x^2 + e^{\sin(x)}.$$

```
>function f(x) := 2*x^2+exp(sin(x)) // fungsi numerik
>f(0), f(1), f(pi)
```

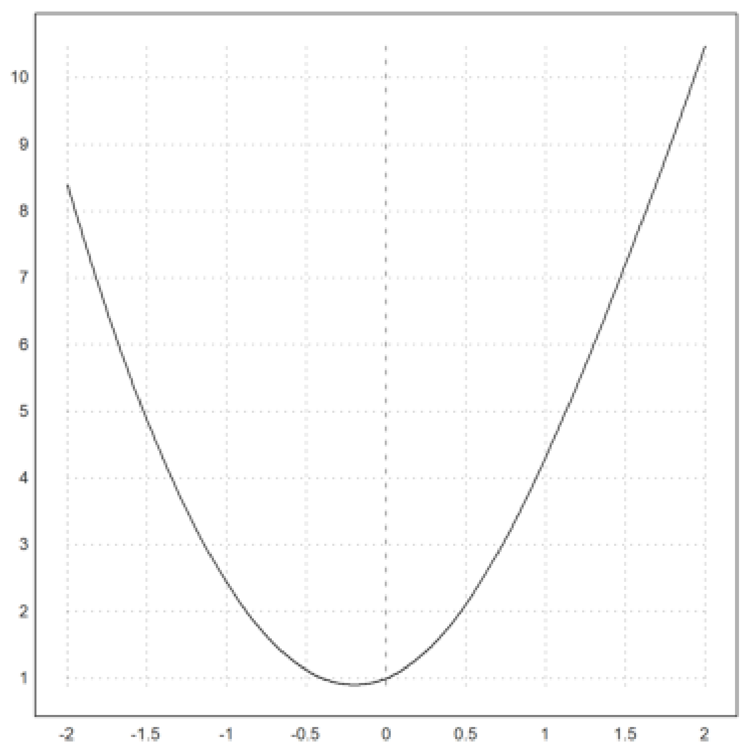
```
1
4.31977682472
20.7392088022
```

```
>f(a) // tidak dapat dihitung nilainya
```

```
Variable or function a not found.
Error in:
f(a) // tidak dapat dihitung nilainya ...
   ^
```

Silakan Anda plot kurva fungsi di atas!

```
>plot2d("f"):
```



Berikutnya kita definisikan fungsi:

$$g(x) = \frac{\sqrt{x^2 - 3x}}{x + 1}.$$

```
>function g(x) := sqrt(x^2-3*x)/(x+1)
>g(3)
```

```
0
```

```
>g(0)
```

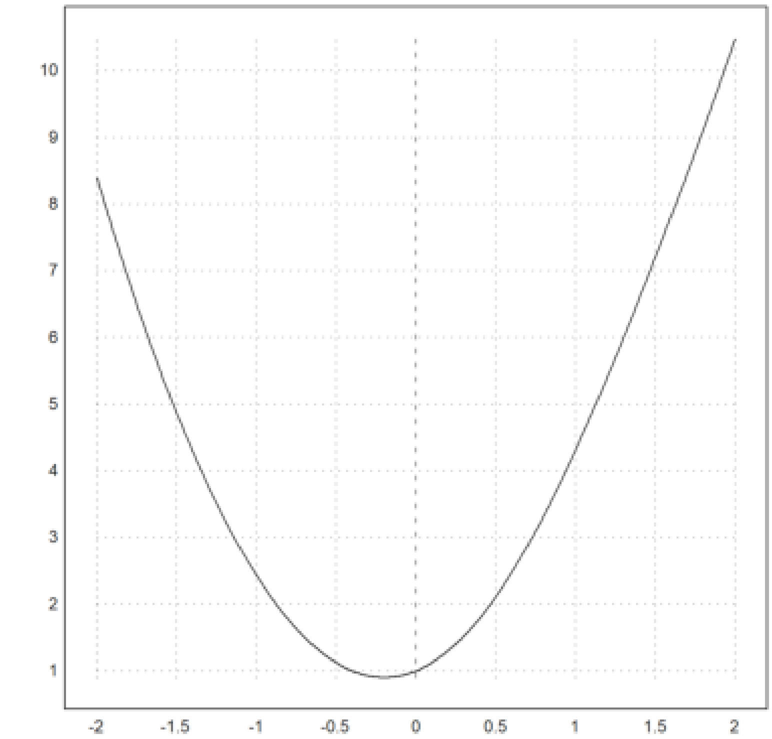
```
0
```

```
>g(1) // kompleks, tidak dapat dihitung oleh fungsi numerik
```

```
Floating point error!
Error in sqrt
Try "trace errors" to inspect local variables after errors.
g:
    useglobal; return sqrt(x^2-3*x)/(x+1)
Error in:
g(1) // kompleks, tidak dapat dihitung oleh fungsi numerik ...
    ^
```

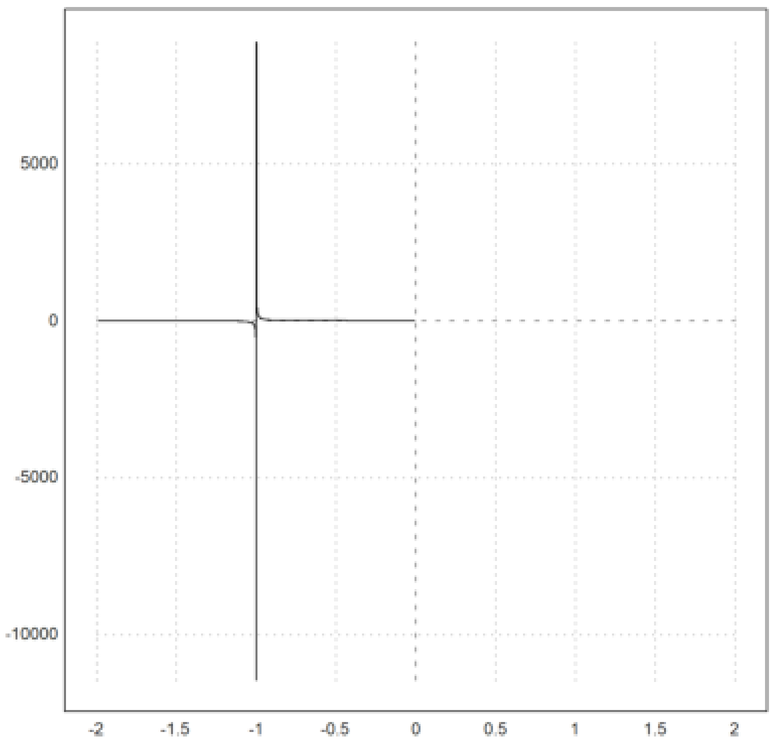
Silakan Anda plot kurva fungsi di atas!
Plot kurva fungsi di atas adalah sebagai berikut.

```
>plot2d("f"):
```



10
9
8
7
6
5
4
3
2
1
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2


```
>plot2d("g"):
```



5000
0
-5000
-10000
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2

```
>f(g(5)) // komposisi fungsi
```

```
2.20920171961
```

```
>g(f(5))
```

```
0.950898070639
```

```
>function h(x) := f(g(x)) // definisi komposisi fungsi
>h(5) // sama dengan f(g(5))
```

```
2.20920171961
```

Silakan Anda plot kurva fungsi komposisi fungsi f dan g:

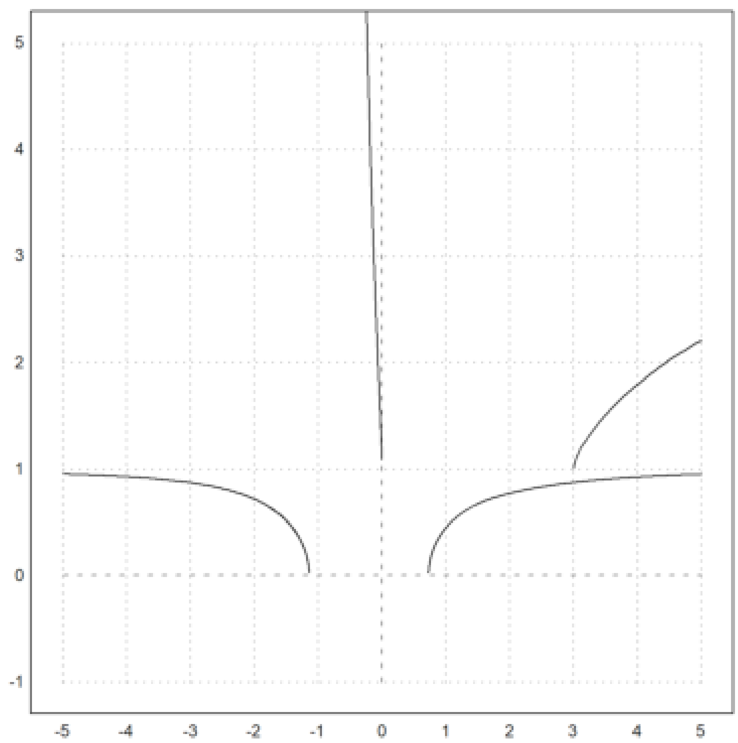
$$h(x) = f(g(x))$$

dan

$$u(x) = g(f(x))$$

bersama-sama kurva fungsi f dan g dalam satu bidang koordinat.

```
>function u(x):= g(f(x));
>plot2d("h",a=-5,b=5,c=-1,d=5); plot2d("u",>add):
```




```
>f(0:10) // nilai-nilai f(0), f(1), f(2), ..., f(10)
```

```
[1,  4.31978,  10.4826,  19.1516,  32.4692,  50.3833,  72.7562,
99.929,  130.69,  163.51,  200.58]
```

```
>fmap(0:10) // sama dengan f(0:10), berlaku untuk semua fungsi
```

```
[1,  4.31978,  10.4826,  19.1516,  32.4692,  50.3833,  72.7562,
99.929,  130.69,  163.51,  200.58]
```

```
>gmap(200:210)
```

```
[0.987534,  0.987596,  0.987657,  0.987718,  0.987778,  0.987837,
0.987896,  0.987954,  0.988012,  0.988069,  0.988126]
```

Misalkan kita akan mendefinisikan fungsi

$$f(x) = \begin{cases} x^3 & x > 0 \\ x^2 & x \le 0. \end{cases}$$

Fungsi tersebut tidak dapat didefinisikan sebagai fungsi numerik secara "inline" menggunakan format :=, melainkan didefinisikan sebagai program. Perhatikan, kata "map" digunakan agar fungsi dapat menerima vektor sebagai input, dan hasilnya berupa vektor. Jika tanpa kata "map" fungsinya hanya dapat menerima input satu nilai.

```
>function map f(x) ...
```

```
    if x>0 then return x^3
    else return x^2
    endif;
  endfunction
```

```
>f(1)
```

```
  1
```

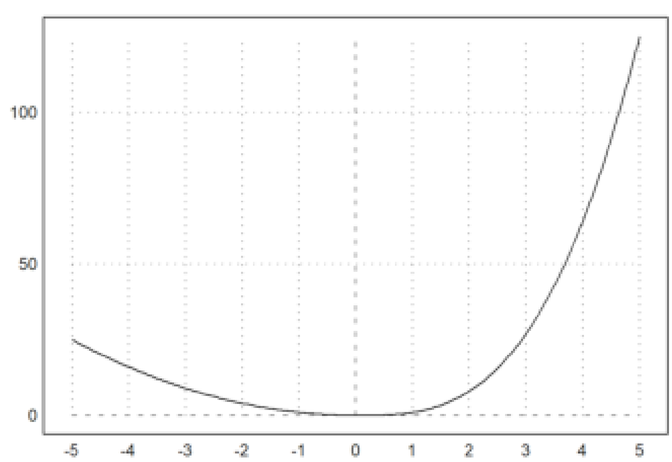
```
>f(-2)
```

```
  4
```

```
>f(-5:5)
```

```
  [25,   16,   9,   4,   1,   0,   1,   8,   27,   64,   125]
```

```
>aspect(1.5); plot2d("f(x)",-5,5):
```



```
>function f(x) &= 2*E^x // fungsi simbolik
```

```
                                   x
                                2 E
```

```
>$f(a) // nilai fungsi secara simbolik
```

$$2\,e^{a}$$

```
>f(E) // nilai fungsi berupa bilangan desimal
```

```
30.308524483
```

```
>$f(E), $float(%)
```

$$2\,e^{e}$$

$$30.30852448295852$$

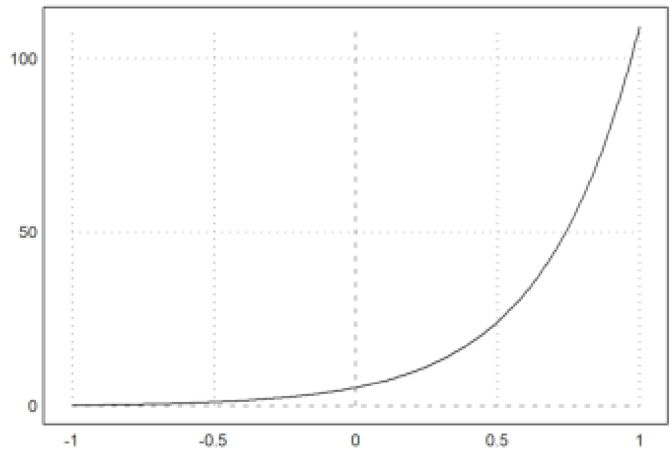
```
>function g(x) &= 3*x+1
```

```
                              3 x + 1
```

```
>function h(x) &= f(g(x)) // komposisi fungsi
```

```
                                3 x + 1
                             2 E
```

```
>plot2d("h(x)",-1,1):
```



Latihan

Bukalah buku Kalkulus. Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan fungsi-fungsi tersebut dan komposisinya di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi). Untuk setiap fungsi, hitung beberapa nilainya, baik untuk satu nilai maupun vektor. Gambar grafik fungsi-fungsi tersebut dan komposisi-komposisi 2 fungsi.

Juga, carilah fungsi beberapa (dua) variabel. Lakukan hal sama seperti di atas.
Fungsi 1

$$a(x) = 2x + 5$$

```
>function a(x):= 2*x+5
>a(0), a(1)
```

```
5
7
```

```
>a(-2:5)
```

```
[1,  3,  5,  7,  9,  11,  13,  15]
```

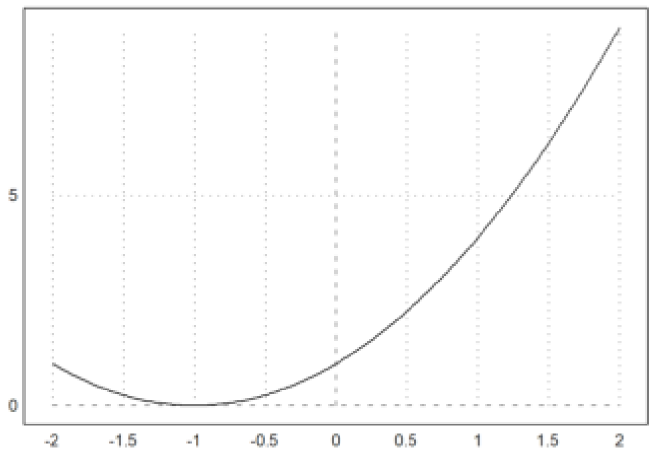
Fungsi 2

$$b(x) = x^2 + 2x + 1$$

```
>function b(x):= x^2+2*x+1
>b(1:10)
```

```
[4,  9,  16,  25,  36,  49,  64,  81,  100,  121]
```

```
>plot2d("b"):
```



Fungsi 3

$$c(x) = \sqrt{(}x - 1)$$

```
>function c(x):= sqrt(x-1)
>cmap(1:3)
```

```
[0,  1,  1.41421]
```

```
>c(101)
```

```
10
```

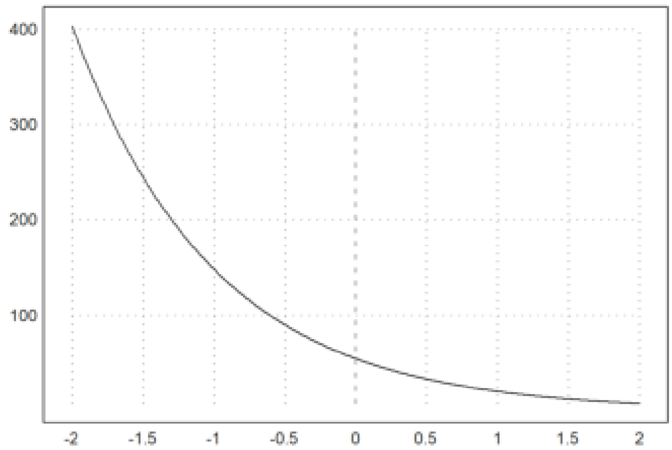
Fungsi 4

$$d(x) = \frac{1}{exp(x-4)}$$

```
>function d(x):= 1/exp(x-4)
>dmap(1:12)
```

```
[20.0855,  7.38906,  2.71828,  1,  0.367879,  0.135335,  0.0497871,
0.0183156,  0.00673795,  0.00247875,  0.000911882,  0.000335463]
```

```
>plot2d("d"):
```



Fungsi 5

$$e(x) = \sin(x) - 2$$

```
>function e(x):= sin(x)-2
>e(2*pi)
```

```
-2
```

```
>e(pi/2)
```

```
-1
```

```
>e(0:pi)
```

```
[-2,  -1.15853,  -1.0907,  -1.85888]
```

Fungsi 6

$$f(x) = a(b(x))$$

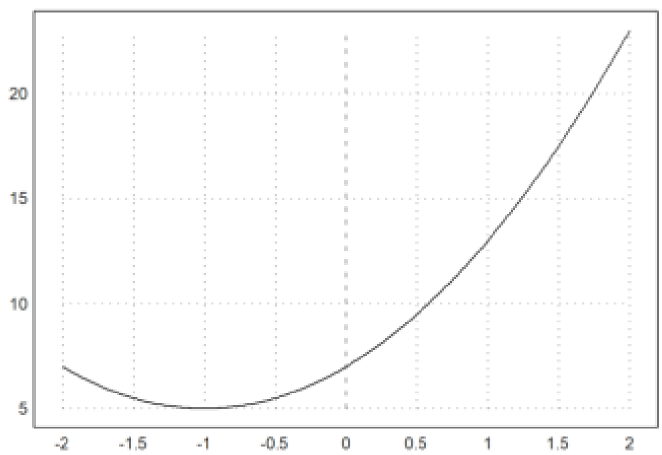
```
>function f(x):= a(b(x))
>f(100)
```

```
20407
```

```
>f(1:9)
```

```
[13,  23,  37,  55,  77,  103,  133,  167,  205]
```

```
>plot2d("f"):
```



Fungsi 7

$$g(x) = c(d(x))$$

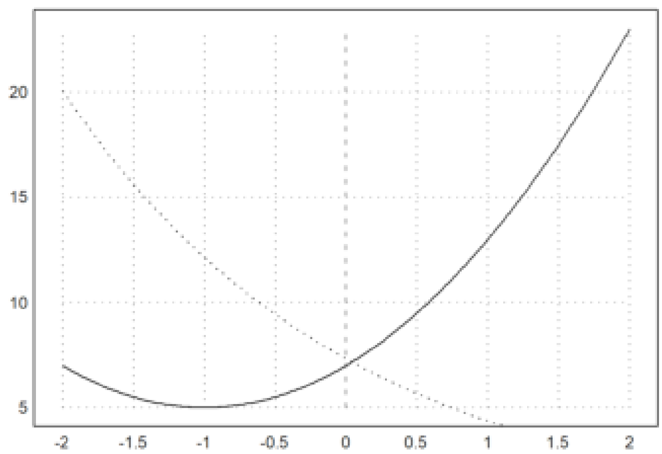
```
>function g(x):= c(d(x))
>g(0)
```

7.32107574289

```
>g(2)
```

2.52765822431

```
>plot2d("f"); plot2d("g",style=".",>add):
```



Fungsi 8

$$j(x)=\begin{cases}x^2-2, & \text{untuk} x\leq 0\\ x^2-2, & \text{untuk} xyanglain\end{cases}$$

```
>function map j(x) ...
```

```
    if x<=-1 then return 2*x+3
    else return x^2-2
    endif;
  endfunction
```

```
>j(1)
```

```
-1
```

```
>j(-5)
```

```
-7
```

Fungsi 9

$$k(x,y) = \sqrt{1-(x^2+y^2)}$$

```
>function k(x,y):=sqrt(1-(x^2+y^2))
>k(1,0)
```

```
0
```

```
>k(1,0)
```

```
0
```

Fungsi 10

$$m(x,y) = x^3 - 2xy + 3y$$

```
>function m(x,y):= x^3-2*x*y+3*y
>m(-2,3)
```

```
13
```

```
>m(1/9:2,2:3)
```

```
[5.55693,  3.70508]
```

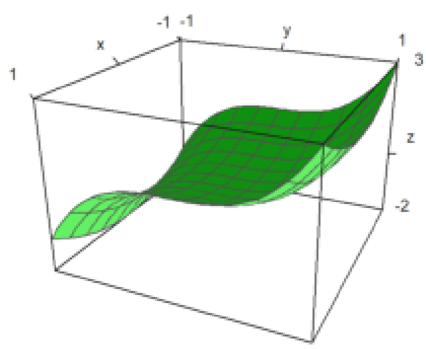
Fungsi 11

$$n(x,y)=x^2+2y^3$$

```
>function n(x,y):= x^2+2*y^3
>n(3,-1)
```

7

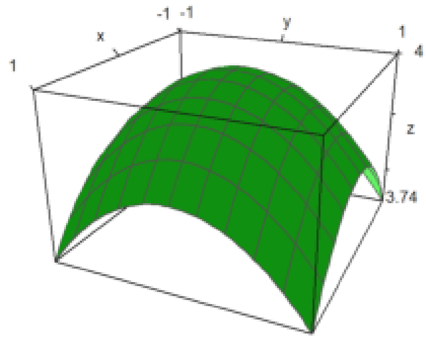
```
>plot3d("n"):
```



Fungsi 12

$$p(x,y)=\sqrt{16-(x^2+y^2)}$$

```
>function p(x,y):=sqrt(16-(x^2+y^2))
>plot3d("p",>user):
```



Fungsi 13

$$q(x,y) = \frac{x}{y} + xy$$

```
>function q(x,y):= x/y + x*y
>q(1,2)
```

```
2.5
```

```
>q(-3,-9)
```

```
27.3333333333
```

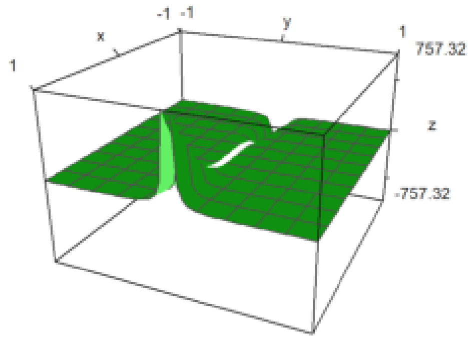
Fungsi 14

$$r(x,y) = (\frac{x^2}{y^2})\sin(x)$$

```
>function r(x,y):= (x^2/y^2)*sin(x)
>r(pi:2*pi,pi)
```

```
[0,  -1.46243,  -2.43558,  -0.539325]
```

```
>plot3d("r"):
```



Menghitung Limit

Perhitungan limit pada EMT dapat dilakukan dengan menggunakan fungsi Maxima, yakni "limit". Fungsi "limit" dapat digunakan untuk menghitung limit fungsi dalam bentuk ekspresi maupun fungsi yang sudah didefinisikan sebelumnya. Nilai limit dapat dihitung pada sebarang nilai atau pada tak hingga (-inf, minf, dan inf). Limit kiri dan limit kanan juga dapat dihitung, dengan cara memberi opsi "plus" atau "minus". Hasil limit dapat berupa nilai, "und" (tak definisi), "ind" (tak tentu namun terbatas), "infinity" (kompleks tak hingga).

Perhatikan beberapa contoh berikut. Perhatikan cara menampilkan perhitungan secara lengkap, tidak hanya menampilkan hasilnya saja.

```
>$showev('limit(sqrt(x^2-3*x)/(x+1),x,inf))
```

$$\lim_{x\rightarrow\infty}{\frac{\sqrt{x^2-3\,x}}{x+1}}=1$$

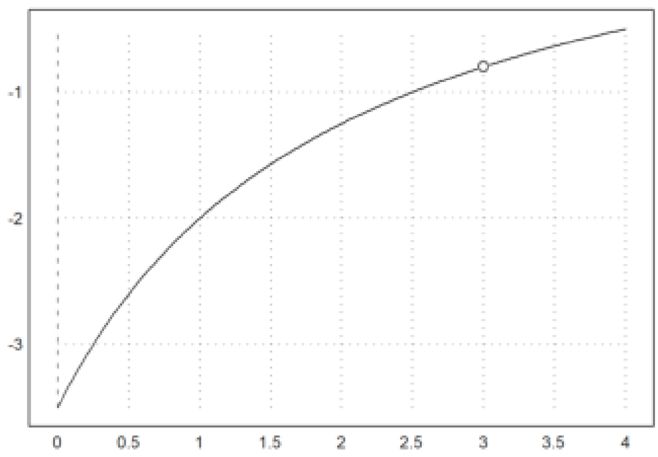
```
>$limit((x^3-13*x^2+51*x-63)/(x^3-4*x^2-3*x+18),x,3)
```

$$-\frac{4}{5}$$

$$\lim_{x\rightarrow 3}{\frac{x^3-13\,x^2+51\,x-63}{x^3-4\,x^2-3\,x+18}}=-\frac{4}{5}$$

Fungsi tersebut diskontinu di titik x=3. Berikut adalah grafik fungsinya.

```
>aspect(1.5); plot2d("(x^3-13*x^2+51*x-63)/(x^3-4*x^2-3*x+18)",0,4); plot2d
```



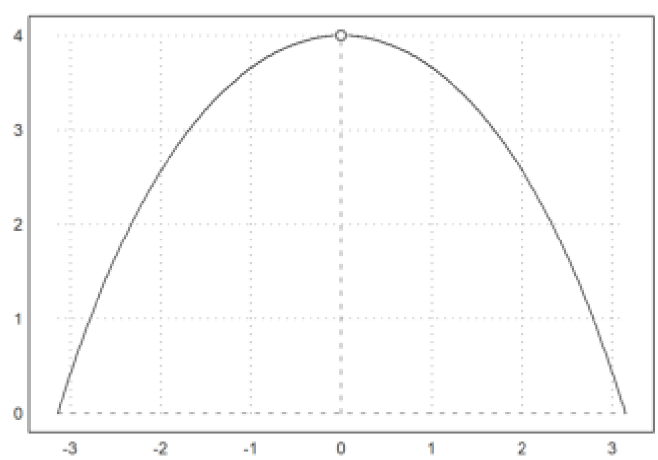
```
>$limit(2*x*sin(x)/(1-cos(x)),x,0)
```

$$4$$

$$2\left(\lim_{x\rightarrow 0}{\frac{x\,\sin x}{1-\cos x}}\right)=4$$

Fungsi tersebut diskontinu di titik x=0. Berikut adalah grafik fungsinya.

```
>plot2d("2*x*sin(x)/(1-cos(x))",-pi,pi); plot2d(0,4,>points,style="ow",>add
```



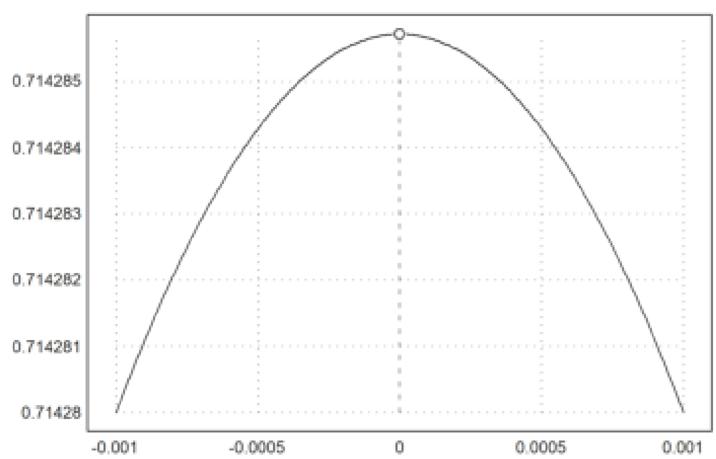
```
>$limit(cot(7*h)/cot(5*h),h,0)
```

$$\frac{5}{7}$$

$$\lim_{h\to 0}{\frac{\cot\left(7\,h\right)}{\cot\left(5\,h\right)}}=\frac{5}{7}$$

Fungsi tersebut juga diskontinu (karena tidak terdefinisi) di x=0. Berikut adalah grafiknya.

```
>plot2d("cot(7*x)/cot(5*x)",-0.001,0.001); plot2d(0,5/7,>points,style="ow",
```

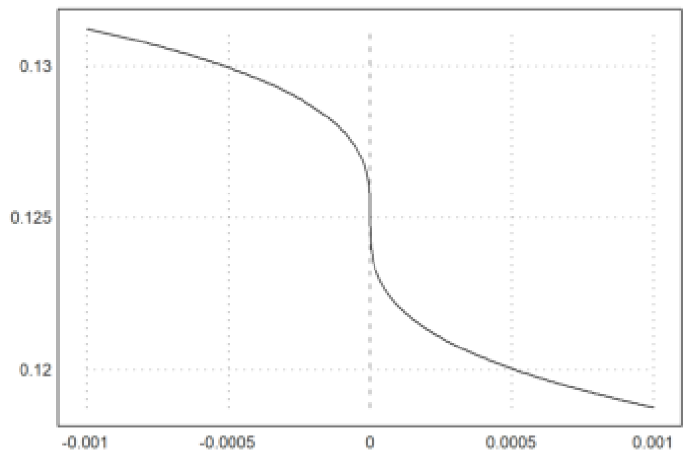



```
>$showev('limit(((x/8)^(1/3)-1)/(x-8),x,8))
```

$$\lim_{x\to 8}{\frac{\frac{x^{\frac{1}{3}}}{2}-1}{x-8}}=\frac{1}{24}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>plot2d("(((x/8)^(1/3))-1)/(x-8)",-0.001,0.001); plot2d(8,1/24,>points,styl
```

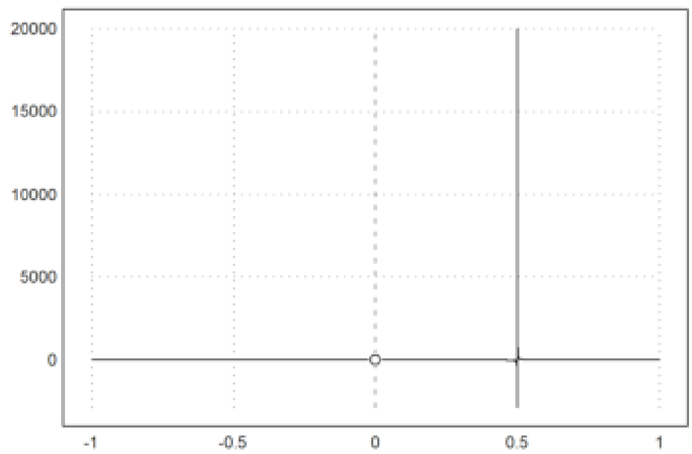



```
>$showev('limit(1/(2*x-1),x,0))
```

$$\lim_{x\to 0}\frac{1}{2\,x-1}=-1$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>plot2d("(1/(2*x-1))",-1,1); plot2d(0,-1,>points,style="ow",>add):
```

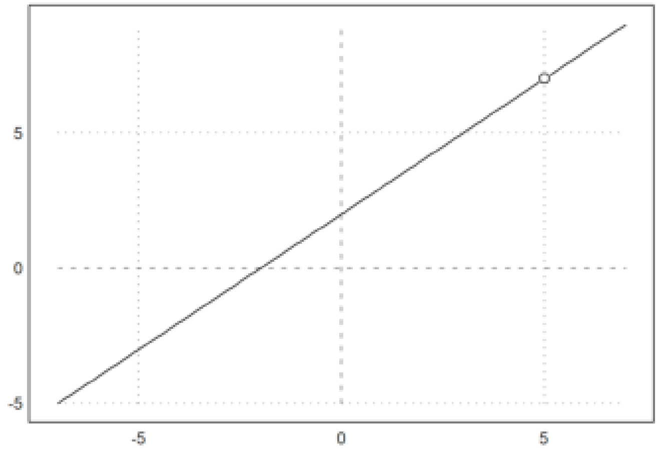



```
>$showev('limit((x^2-3*x-10)/(x-5),x,5))
```

$$\lim_{x\to 5}\frac{x^2-3\,x-10}{x-5}=7$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>plot2d("((x^2-3*x-10)/(x-5))",-7,7); plot2d(5,7,>points,style="ow",>add):
```

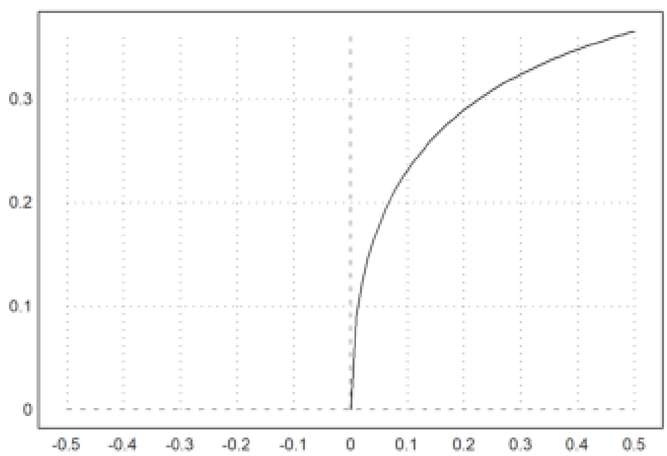



```
>$showev('limit(sqrt(x^2+x)-x,x,inf))
```

$$\lim_{x\to\infty} \sqrt{x^2+x} - x = \frac{1}{2}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

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



```
>$showev('limit(abs(x-1)/(x-1),x,1,minus))
```

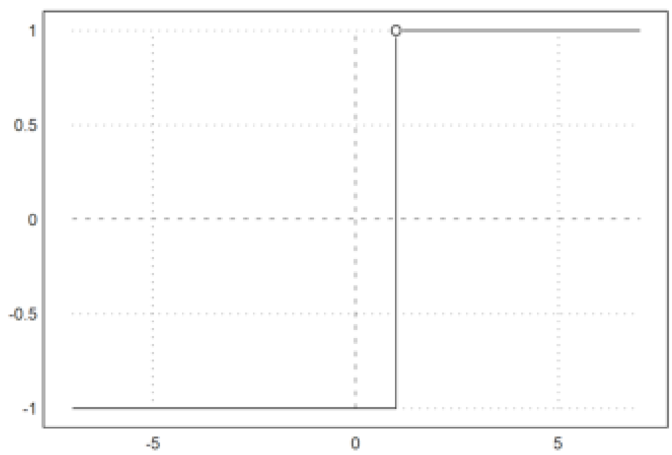
$$\lim_{x\uparrow 1} \frac{|x-1|}{x-1} = -1$$

Hitung limit di atas untuk x menuju 1 dari kanan.
Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>$limit((abs(x-1)/(x-1),x,1))
```

$$1$$

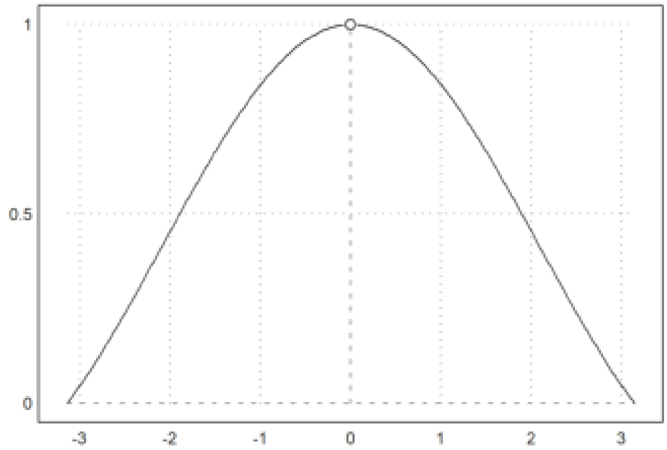
```
>plot2d("(abs(x-1)/(x-1))",-7,7); plot2d(1,1,>points,style="ow",>add):
```




```
>$showev('limit(sin(x)/x,x,0))
```

$$\lim_{x\to 0} \frac{\sin x}{x} = 1$$

```
>plot2d("sin(x)/x",-pi,pi); plot2d(0,1,>points,style="ow",>add):
```

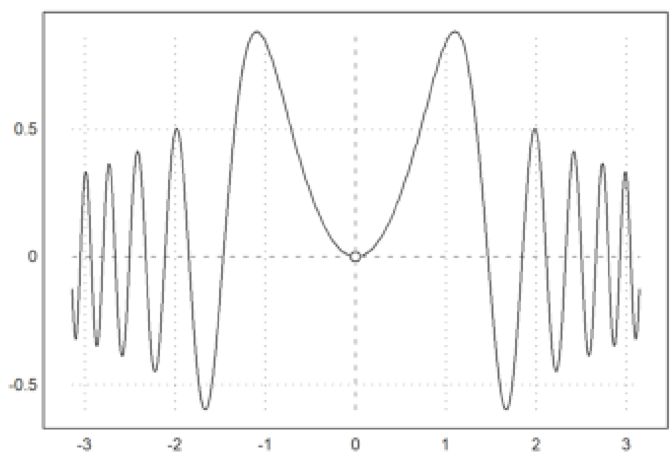



```
>$showev('limit(sin(x^3)/x,x,0))
```

$$\lim_{x\to 0}\frac{\sin x^3}{x}=0$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

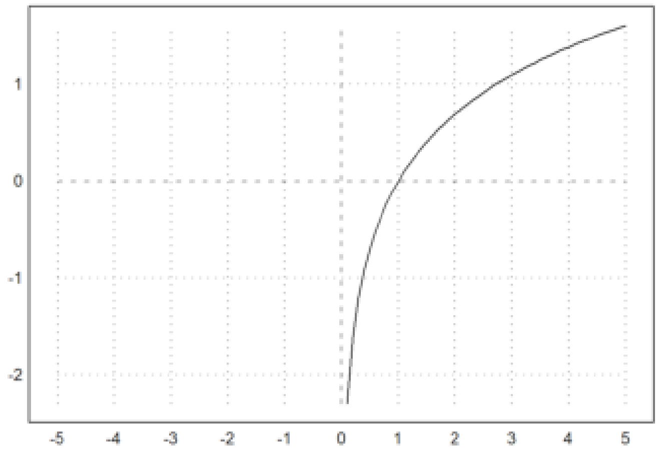
```
>plot2d("sin(x^3)/x",-pi,pi); plot2d(0,0,>points,style="ow",>add):
```




```
>$showev('limit(log(x), x, minf))
```

$$\lim_{x\to -\infty}\log x=\mathit{infinity}$$

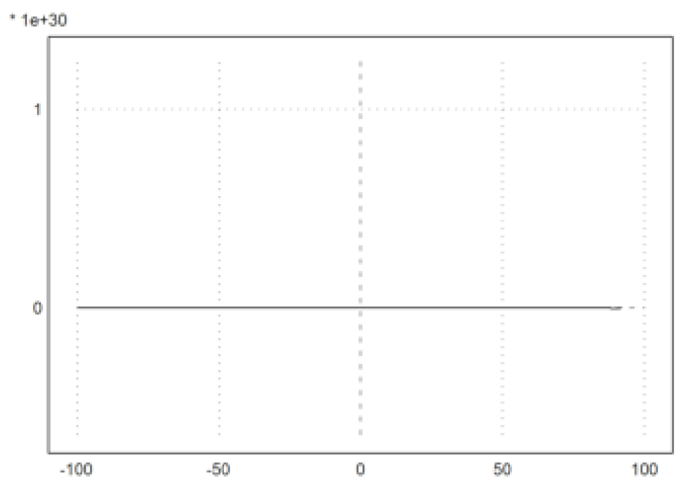
```
>plot2d("log(x)",-5,5):
```




```
>$showev('limit((-2)^x,x, inf))
```

$$\lim_{x\to\infty} (-2)^x = infinity$$

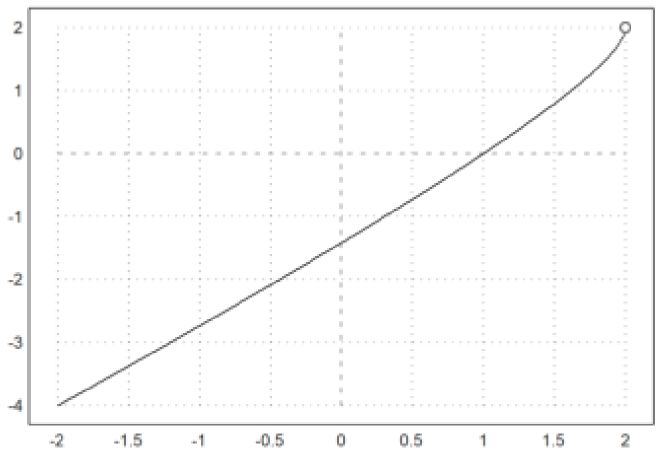
```
>plot2d("(-2)^x",-100,100):
```




```
>$showev('limit(t-sqrt(2-t),t,2,minus))
```

$$\lim_{t \uparrow 2} t - \sqrt{2 - t} = 2$$

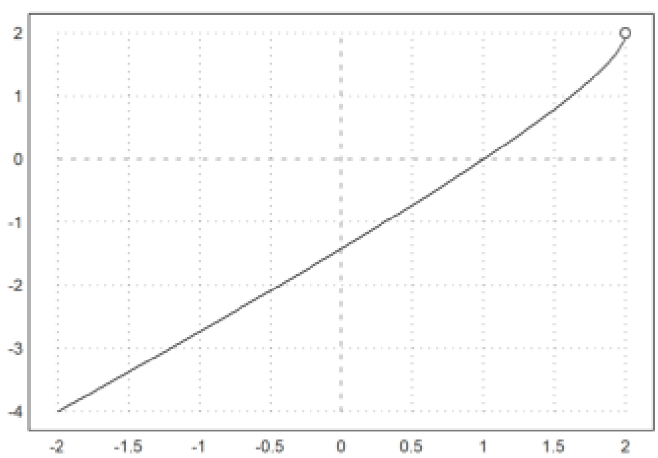
```
>plot2d("x-sqrt(2-x)",-2,2); plot2d(2,2,>points,style="ow",>add):
```




```
>$showev('limit(t-sqrt(2-t),t,2,plus))
```

$$\lim_{t \downarrow 2} t - \sqrt{2 - t} = 2$$

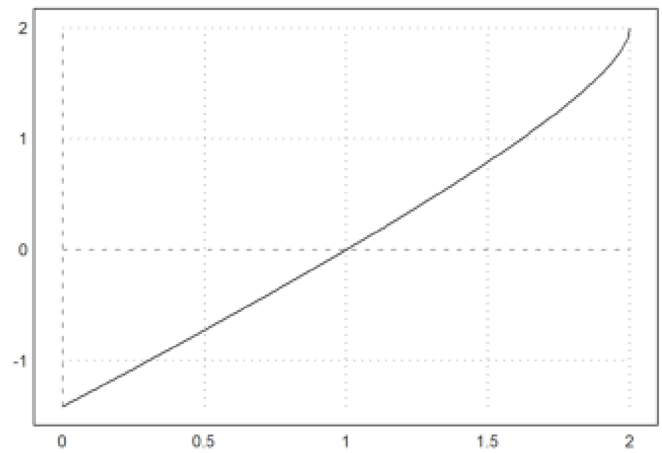
```
>plot2d("x-sqrt(2-x)",-2,2); plot2d(2,2,>points,style="ow",>add):
```



```
>$showev('limit(t-sqrt(2-t),t,5,plus)) // Perhatikan hasilnya
```

$$\lim_{t\downarrow 5} t-\sqrt{2-t}=5-\sqrt{3}\,i$$

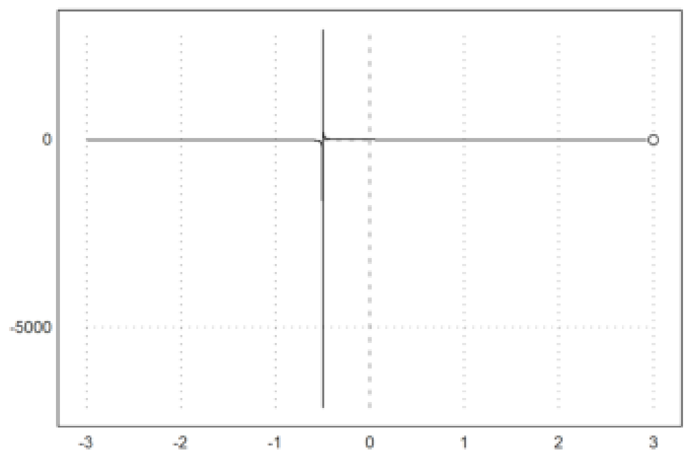
```
>plot2d("x-sqrt(2-x)",0,2):
```




```
>$showev('limit((x^2-9)/(2*x^2-5*x-3),x,3))
```

$$\lim_{x\rightarrow 3}\frac{x^2-9}{2\,x^2-5\,x-3}=\frac{6}{7}$$

```
>plot2d("(x^2-9)/(2*x^2-5*x-3)",-3,3); plot2d(3,6/7,>points,style="ow",>add
```

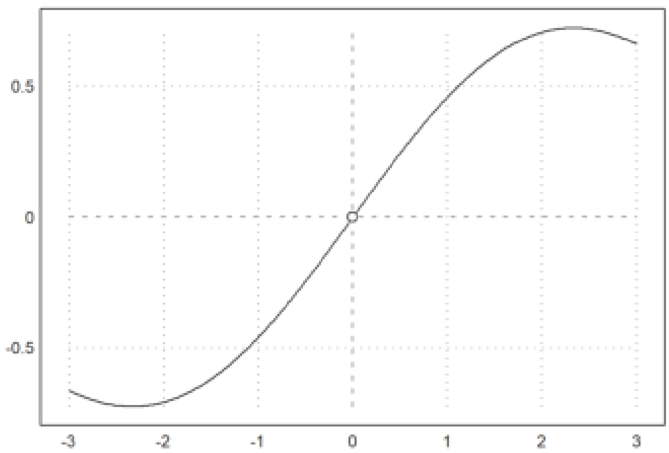


Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>$showev('limit((1-cos(x))/x,x,0))
```

$$\lim_{x \to 0} \frac{1 - \cos x}{x} = 0$$

```
>plot2d("(1-cos(x))/x",-3,3); plot2d(0,0,>points,style="ow",>add):
```

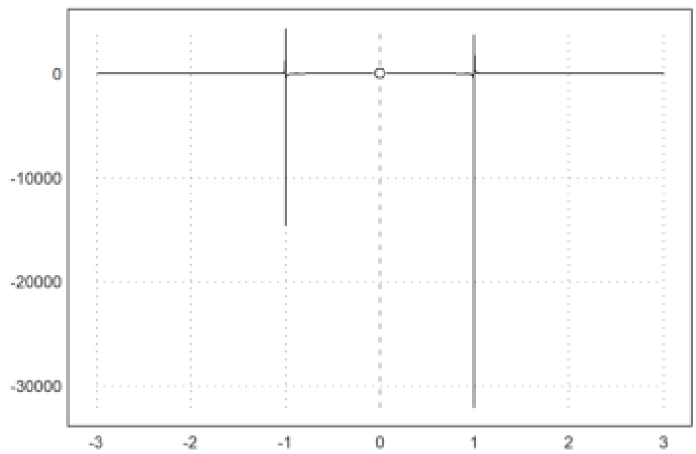


Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>$showev('limit((x^2+abs(x))/(x^2-abs(x)),x,0))
```

$$\lim_{x \to 0} \frac{|x| + x^2}{x^2 - |x|} = -1$$

```
>plot2d("(x^2+abs(x))/(x^2-abs(x))",-3,3); plot2d(0,-1,>points,style="ow",>
```



Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>$showev('limit((1+1/x)^x,x,inf))
```

$$\lim_{x \to \infty} \left(\frac{1}{x} + 1\right)^x = e$$

```
>plot2d("(1+1/x)^x",0,1000):
```

images/EMT4Kalkulus_Wahyu Rananda Westri_22305144039_M

```
>$showev('limit((1+k/x)^x,x,inf))
```

$$\lim_{x \to \infty} \left(\frac{k}{x} + 1\right)^x = e^k$$

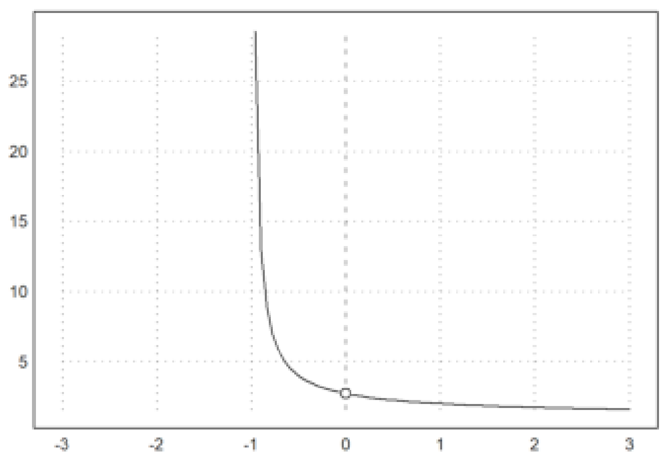
```
>plot2d("(1+k/x)^x",inf,exp(k)):
```

```
Variable or function inf not found.
Error in:
plot2d("(1+k/x)^x",inf,exp(k)): ...
                        ^
```

```
>$showev('limit((1+x)^(1/x),x,0))
```

$$\lim_{x\to 0}{\left(x+1\right)^{\frac{1}{x}}}=e$$

```
>plot2d("(1+x)^(1/x)",-3,3); plot2d(0,E,>points,style="ow",>add):
```



Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>$showev('limit((x/(x+k))^x,x,inf))
```

$$\lim_{x\to \infty }{\left(\frac{x}{x+k}\right)^{x}}=e^ {- k }$$

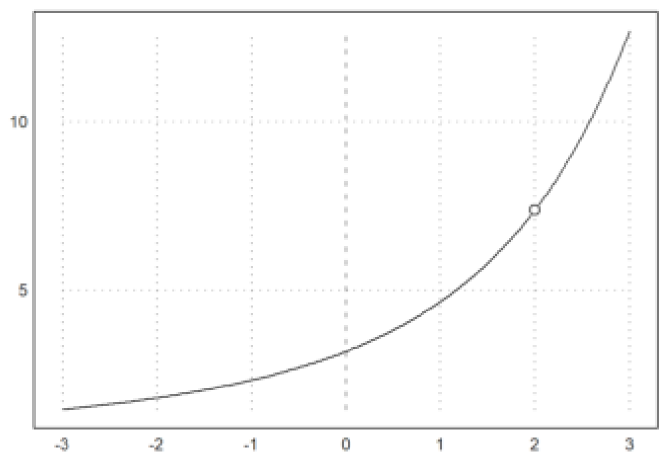
```
>plot2d("(x/(x+k))^x",x,1000):
```

```
Variable or function x not found.
Error in:
plot2d("(x/(x+k))^x",x,1000): ...
                             ^
```

```
>$showev('limit((E^x-E^2)/(x-2),x,2))
```

$$\lim_{x\to 2}\frac{e^x-e^2}{x-2}=e^2$$

```
>plot2d("(E^x-E^2)/(x-2)",-3,3); plot2d(2,exp(2),>points,style="ow",>add):
```

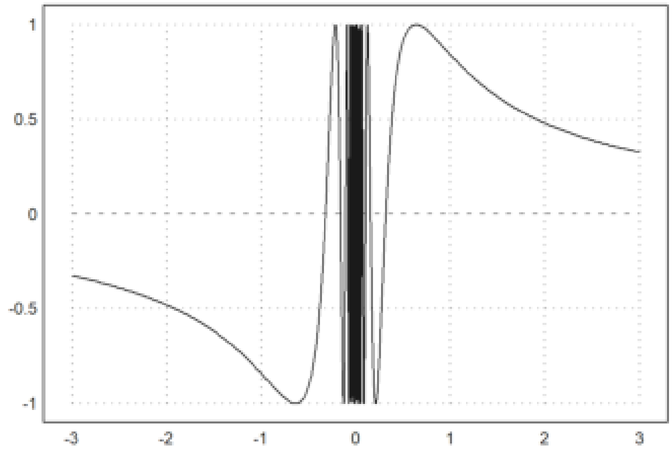


Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>$showev('limit(sin(1/x),x,0))
```

$$\lim_{x\to 0}\sin\left(\frac{1}{x}\right)=ind$$

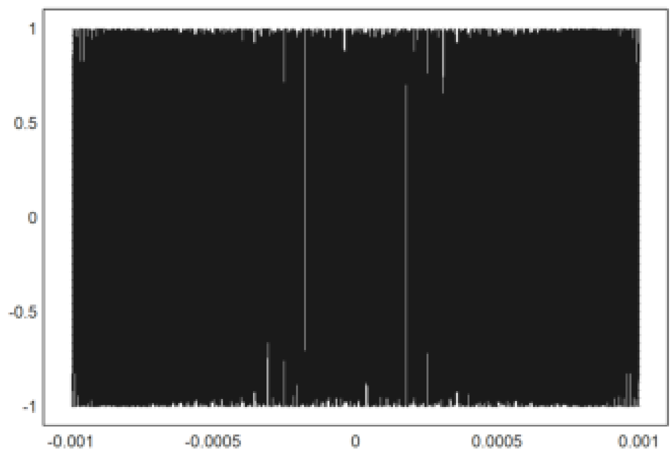
```
>plot2d("sin(1/x)",-3,3):
```




```
>$showev('limit(sin(1/x),x,inf))
```

$$\lim_{x\to\infty} \sin\left(\frac{1}{x}\right) = 0$$

```
>plot2d("sin(1/x)",-0.001,0.001):
```



Latihan

Bukalah buku Kalkulus. Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi). Untuk setiap fungsi, hitung nilai limit fungsi tersebut di beberapa nilai dan di tak hingga. Gambar grafik fungsi tersebut untuk mengkonfirmasi nilai-nilai limit tersebut.

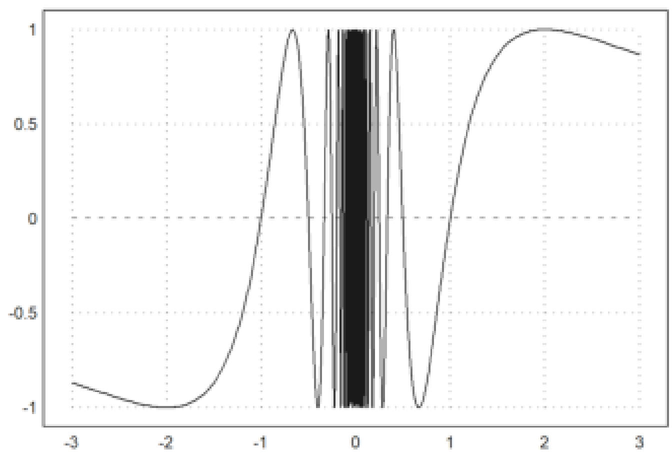
Fungsi 1

$$f(x) = \sin(\pi/x)$$

```
>$showev('limit(sin(pi/x),x,0))
```

$$\lim_{x\to 0}\sin\left(\frac{\pi}{x}\right) = ind$$

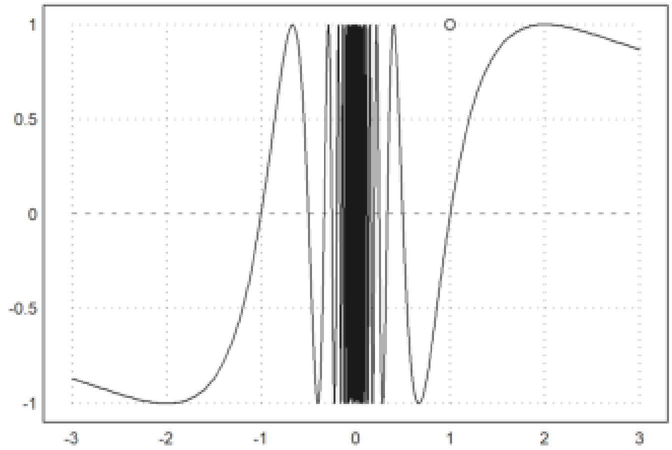
```
>plot2d("(sin(pi/x))",-3,3):
```




```
>$limit((sin(pi/x),x,1))
```

$$1$$

```
>plot2d("(sin(pi/x))",-3,3); plot2d(1,1,>points,style="ow",>add):
```




```
>$limit((sin(pi/x),x,inf))
```

$$\infty$$

Fungsi 2

$$f(x) = \frac{\sin(3x)}{x}$$

```
>$showev('limit((sin(3*x)/x),x,0))
```

$$\lim_{x \to 0} \frac{\sin\left(3\,x\right)}{x} = 3$$

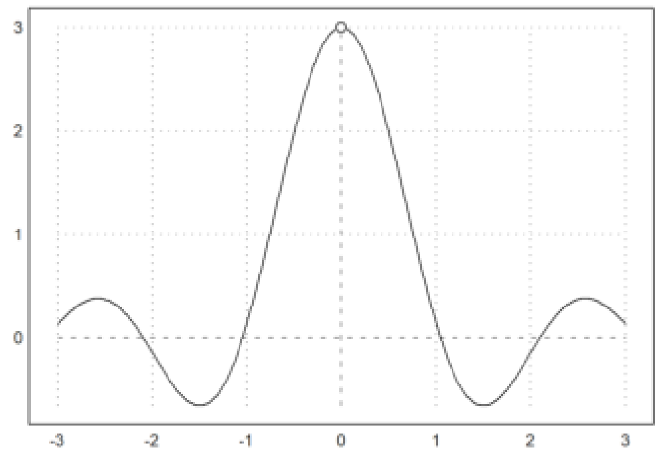
```
>$limit((sin(3*x)/x,x,inf))
```

$$\infty$$

```
>$limit((sin(3*x)/x,x,2))
```

$$2$$

```
>plot2d("(sin(3*x)/x)",-3,3); plot2d(0,3,>points,style="ow",>add):
```



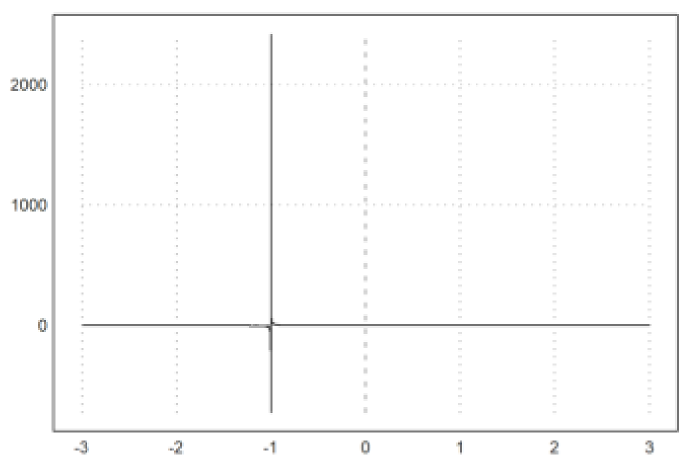
Fungsi 3

$$\frac{x}{x^3 + 1}$$

```
>$showev('limit((x^2/(1+x^3)),x,inf,minus))
```

$$\lim_{x \uparrow \infty} \frac{x^2}{x^3 + 1} = 0$$

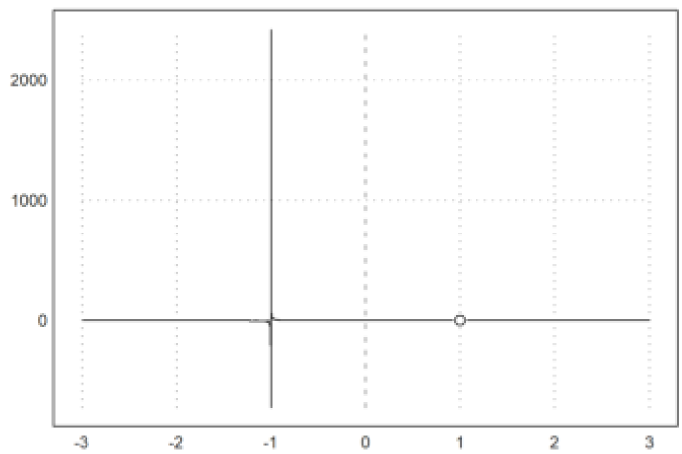
```
>plot2d("(x^2/(1+x^3))",-3,3):
```



```
>$limit((x^2/(1+x^3)),x,1)
```

$$\frac{1}{2}$$

```
>plot2d("(x^2/(1+x^3))",-3,3); plot2d(1,1/2,>points,style="ow",>add):
```



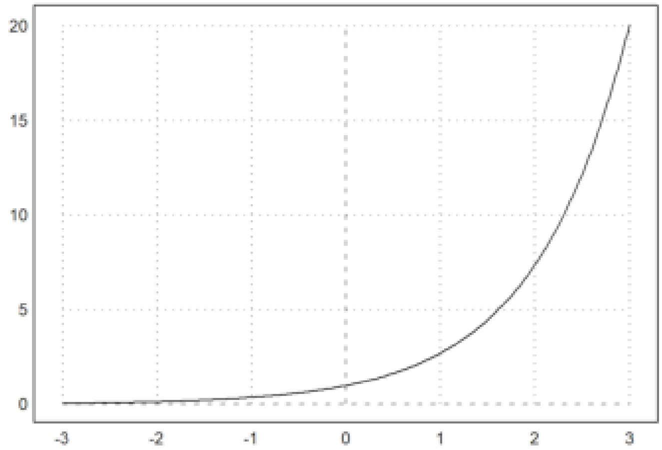
Fungsi 4

$$f(x) = e^x$$

```
>$showev('limit((exp(x)),x,inf,plus))
```

$$\lim_{x\downarrow\infty} e^x = \infty$$

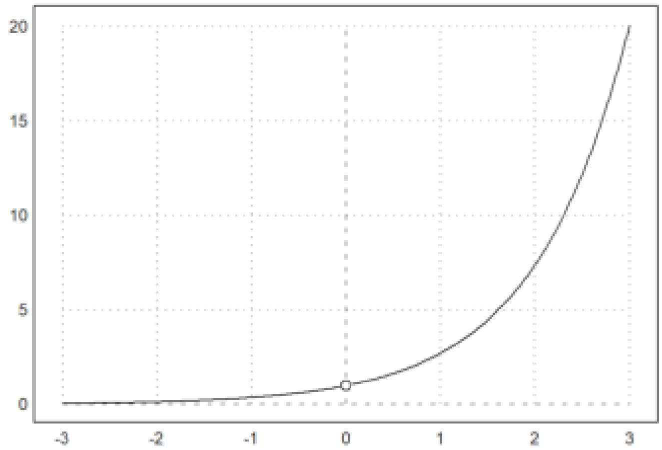
```
>plot2d("exp(x)",-3,3):
```




```
>$limit(exp(x),x,0)
```

$$1$$

```
>plot2d("(exp(x))",-3,3); plot2d(0,1,>points,style="ow",>add):
```



Fungsi 5

$$f(x) = \frac{|x-2|}{x-2}$$

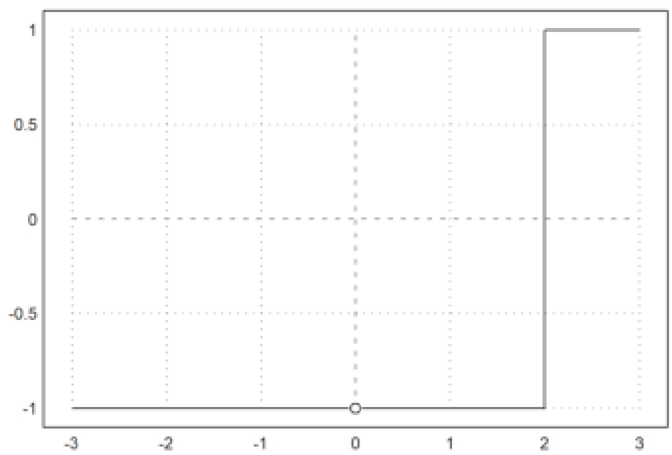
```
>$showev('limit(abs(x-2)/(x-2),x,2,plus))
```

$$\lim_{x\downarrow 2}\frac{|x-2|}{x-2}=1$$

```
>$limit(abs(x-2)/(x-2),x,0)
```

$$-1$$

```
>plot2d("abs(x-2)/(x-2)",-3,3); plot2d(0,-1,>points,style="ow",>add):
```



Turunan Fungsi

Definisi turunan:

$$f'(x)=\lim_{h\to 0}\frac{f(x+h)-f(x)}{h}$$

Berikut adalah contoh-contoh menentukan turunan fungsi dengan menggunakan definisi turunan (limit).

```
>$showev('limit(((x+h)^2-x^2)/h,h,0)) // turunan x^2
```

$$\lim_{h\to 0}\frac{(x+h)^2-x^2}{h}=2\,x$$

```
>p &= expand((x+h)^2-x^2)|simplify; $p //pembilang dijabarkan dan disederha
```

$$2\,h\,x+h^2$$

```
>q &=ratsimp(p/h); $q // ekspresi yang akan dihitung limitnya disederhanaka
```

$$2\,x+h$$

```
>$limit(q,h,0) // nilai limit sebagai turunan
```

$$2\,x$$

```
>$showev('limit(((x+h)^n-x^n)/h,h,0)) // turunan x^n
```

$$\lim_{h\to 0}\frac{(x+h)^n-x^n}{h}=n\,x^{n-1}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, ekspansikan (x+h)^n dengan menggunakan teorema binomial.

BUKTI

$$f'(x)=\lim_{h\to 0}\frac{f(x+h)-f(x)}{h}$$

Untuk

$$f(x) = x^n$$

$$\frac{d}{dx}sin(x) = \lim_{h\to 0} \frac{(x+h)^n - x^n}{h}$$

Dengan

$$(a+b)^n = \sum_{k=0}^{n} a^k b^{n-k}$$

maka

$$= \lim_{h\to 0} \frac{(x^n + \frac{n}{1!}x^{n-1}h + \frac{n(n-1)}{2!}x^{n-2}h^2 + \frac{n(n-1)(n-2)}{3!}x^{n-3}h^3 + ...) - x^n}{h}$$

$$= \lim_{h\to 0} \frac{n.x^{n-1}h + \frac{n(n-1)}{2!}x^{n-2}h^2 + \frac{n(n-1)(n-2)}{3!}x^{n-3}h^3 + ...}{h}$$

$$= \lim_{h\to 0} n.x^{n-1} + \frac{n(n-1)}{2!}.x^{n-2}h + \frac{n(n-1)(n-2)}{3!}.x^{n-3}h^2 + ...$$

$$= n.x^{n-1} + 0 + 0 + ... + 0$$

$$= n.x^{n-1}$$

Jadi, terbukti benar bahwa

$$f'(x^n) = n.x^{n-1}$$

```
>$showev('limit((sin(x+h)-sin(x))/h,h,0)) // turunan sin(x)
```

$$\lim_{h\to 0} \frac{\sin(x+h) - \sin x}{h} = \cos x$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, ekspansikan sin(x+h) dengan menggunakan rumus jumlah dua sudut.

Bukti

$$f'(x) = \lim_{h\to 0} \frac{sin(x+h) - sin(x)}{h}$$

$$sin(a+b) = sin(a)cos(a) + cos(a)sin(b)$$

$$= \lim_{h\to 0} \frac{sin(x)cos(h) + cos(x)sin(h) - sin(x)}{h}$$

$$= \lim_{h\to 0} sinx.\frac{cos(h)-1}{h} + \lim_{h\to 0} cos(x).\frac{sin(h)}{h}$$

$$= sin(x).0 + cos(x).1$$

$$= cos(x)$$

Jadi, terbukti benar bahwa

$$f'(sin(x)) = cos(x)$$

```
>$showev('limit((log(x+h)-log(x))/h,h,0)) // turunan log(x)
```

$$\lim_{h\to 0} \frac{\log(x+h) - \log x}{h} = \frac{1}{x}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, gunakan sifat-sifat logaritma dan hasil limit pada bagian sebelumnya di atas.

Bukti

$$f'(x) = \lim_{h\to 0} \frac{log(x+h) - logx}{h}$$

$$= \lim_{h\to 0} \frac{\frac{d}{dh}(log(x+h) - logx)}{\frac{d}{dh}(h)}$$

$$= \lim_{h\to 0} \frac{\frac{1}{x+h}}{1}$$

$$= \lim_{h\to 0} \frac{1}{x+h}$$

$$= \frac{1}{x}$$

Jadi, terbukti benar bahwa

$$f'(x) = \lim_{h\to 0} \frac{log(x+h) - logx}{h} = \frac{1}{x}$$

```
>$showev('limit((1/(x+h)-1/x)/h,h,0)) // turunan 1/x
```

$$\lim_{h\to 0} \frac{\frac{1}{x+h} - \frac{1}{x}}{h} = -\frac{1}{x^2}$$

```
>$showev('limit((E^(x+h)-E^x)/h,h,0)) // turunan f(x)=e^x
```

```
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Maxima is asking
Acceptable answers are: yes, y, Y, no, n, N, unknown, uk
Is x an integer?

Use assume!
Error in:
$showev('limit((E^(x+h)-E^x)/h,h,0)) // turunan f(x)=e^x ...
                                      ^
```

Maxima bermasalah dengan limit:

$$\lim_{h\to 0} \frac{e^{x+h}-e^x}{h}.$$

Oleh karena itu diperlukan trik khusus agar hasilnya benar.

```
>$showev('limit((E^h-1)/h,h,0))
```

$$\lim_{h\to 0} \frac{e^h-1}{h} = 1$$

```
>$showev('factor(E^(x+h)-E^x))
```

$$factor\left(e^{x+h}-e^x\right) = \left(e^h-1\right)\,e^x$$

```
>$showev('limit(factor((E^(x+h)-E^x)/h),h,0)) // turunan f(x)=e^x
```

$$\left(\lim_{h\to 0} \frac{e^h-1}{h}\right)\,e^x = e^x$$

```
>function f(x) &= x^x
```

```
                                 x
                                x
```

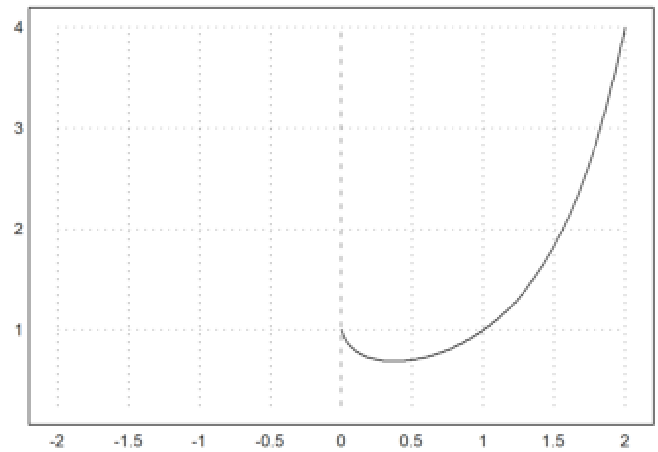
```
>$showev('limit(f(x),x,0))
```

$$\lim_{x\to 0}x^{x}=1$$

Silakan Anda gambar kurva

$$y = x^x.$$

```
>plot2d("x^x"):
```




```
>$showev('limit((f(x+h)-f(x))/h,h,0)) // turunan f(x)=x^x
```

$$\lim_{h\to 0}\frac{\left(x+h\right)^{x+h}-x^{x}}{h}=\mathit{infinity}$$

Di sini Maxima juga bermasalah terkait limit:

$$\lim_{h\to 0}\frac{(x+h)^{x+h}-x^x}{h}.$$

Dalam hal ini diperlukan asumsi nilai x.

```
>&assume(x>0); $showev('limit((f(x+h)-f(x))/h,h,0)) // turunan f(x)=x^x
```

$$\lim_{h\to 0}\frac{\left(x+h\right)^{x+h}-x^{x}}{h}=x^{x}\,\left(\log x+1\right)$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

```
>&forget(x>0) // jangan lupa, lupakan asumsi untuk kembali ke semula
```

```
[x > 0]
```

```
>&forget(x<0)
```

```
[x < 0]
```

```
>&facts()
```

```
[]
```

```
>$showev('limit((asin(x+h)-asin(x))/h,h,0)) // turunan arcsin(x)
```

$$\lim_{h\rightarrow 0}{\frac{\arcsin \left(x+h\right)-\arcsin x}{h}}=\frac{1}{\sqrt{1-x^2}}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

```
>$showev('limit((tan(x+h)-tan(x))/h,h,0)) // turunan tan(x)
```

$$\lim_{h\rightarrow 0}{\frac{\tan \left(x+h\right)-\tan x}{h}}=\frac{1}{\cos ^2x}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

```
>function f(x) &= sinh(x) // definisikan f(x)=sinh(x)
```

```
                              sinh(x)
```

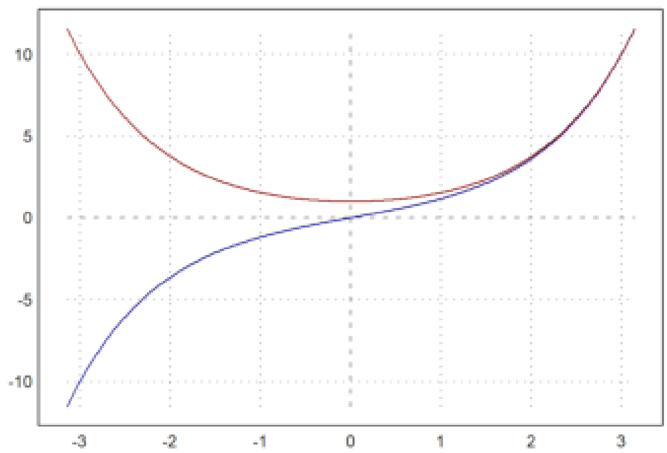
```
>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x) // df(x) = f'(x)
```

$$\frac{e^{-x}\,\left(e^{2\,x}+1\right)}{2}$$

Hasilnya adalah cosh(x), karena

$$\frac{e^x + e^{-x}}{2} = \cosh(x).$$

```
>plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]):
```




```
>function f(x) &= sin(3*x^5+7)^2
```

```
                        2      5
                     sin (3 x   + 7)
```

```
>diff(f,3), diffc(f,3)
```

```
1198.32948904
1198.72863721
```

Apakah perbedaan diff dan diffc?

Diferensiasi numerik pada dasarnya agak tidak akurat untuk fungsi-fungsi umum. Untuk mendapatkan perkiraan yang baik, turunan pertama menggunakan 4 evaluasi fungsi. Ada fungsi diffc() yang lebih akurat untuk fungsi-fungsi yang bersifat analitis dan bernilai riil pada garis bilangan riil.

```
>$showev('diff(f(x),x))
```

$$\frac{d}{d\,x}\,\sin^2\left(3\,x^5+7\right)=30\,x^4\,\cos\left(3\,x^5+7\right)\,\sin\left(3\,x^5+7\right)$$

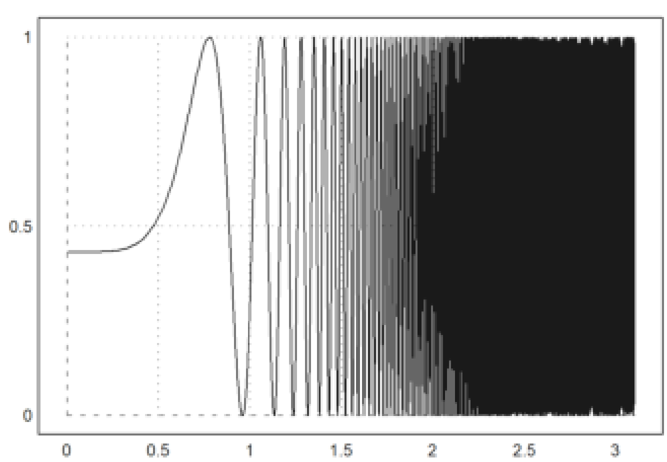
```
>$% with x=3
```

$$\%at\left(\frac{d}{d\,x}\,\sin^2\left(3\,x^5+7\right)\,,x=3\right)=2430\,\cos 736\,\sin 736$$

```
>$float(%)
```

$$\%at\left(\frac{d^{1.0}}{d\,x^{1.0}}\,\sin^2\left(3.0\,x^5+7.0\right)\,,x=3.0\right)=1198.728637211748$$

```
>plot2d(f,0,3.1):
```



```
>function f(x) &=5*cos(2*x)-2*x*sin(2*x) // mendifinisikan fungsi f
```

```
5 cos(2 x) - 2 x sin(2 x)
```

```
>function df(x) &=diff(f(x),x) // fd(x) = f'(x)
```

```
- 12 sin(2 x) - 4 x cos(2 x)
```

```
>$'f(1)=f(1), $float(f(1)), $'f(2)=f(2), $float(f(2)) // nilai f(1) dan f(2
```

$$f\left(1\right)=5\,\cos 2-2\,\sin 2$$

$$-3.899329036387075$$

$$f\left(2\right)=5\,\cos 4-4\,\sin 4$$

$$-0.2410081230863468$$

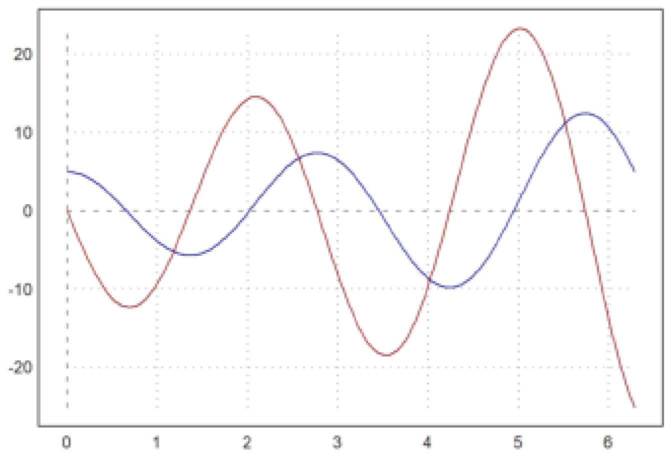
```
>xp=solve("df(x)",1,2,0) // solusi f'(x)=0 pada interval [1, 2]
```

```
1.35822987384
```

```
>df(xp), f(xp) // cek bahwa f'(xp)=0 dan nilai ekstrim di titik tersebut
```

```
0
-5.67530133759
```

```
>plot2d(["f(x)","df(x)"],0,2*pi,color=[blue,red]): //grafik fungsi dan turu
```



Perhatikan titik-titik "puncak" grafik y=f(x) dan nilai turunan pada saat grafik fungsinya mencapai titik "puncak" tersebut.

Latihan

Bukalah buku Kalkulus. Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi). Untuk setiap fungsi, tentukan turunannya dengan menggunakan definisi turunan (limit), menggunakan perintah diff, dan secara manual (langkah demi langkah yang dihitung dengan Maxima) seperti contoh-contoh di atas. Gambar grafik fungsi asli dan fungsi turunannya pada sumbu koordinat yang sama.

Fungsi 1

$$a(x) = \frac{2x^2+1}{1-x^2}$$

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

```
      2
   2 x  + 1
   --------
          2
    1 - x
```

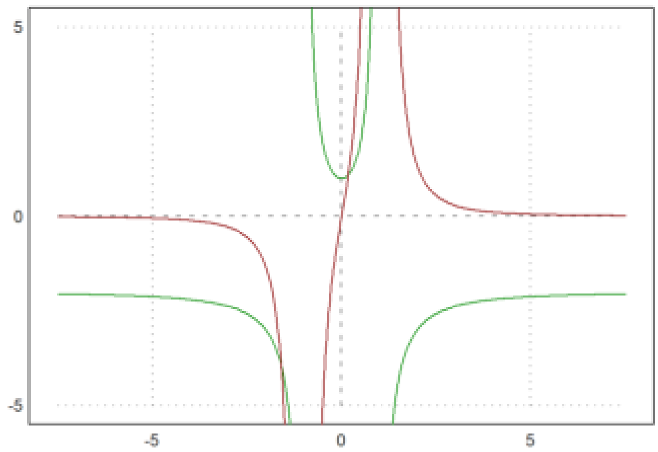
```
>function da(x) &= limit((a(x+h)-a(x))/h,h,0); $da(x) // da(x) = a'(x) meng
```

$$\frac{6\,x}{x^4-2\,x^2+1}$$

```
>$&showev('diff(a(x),x))//turunan menggunakan diff
```

$$\frac{d}{d\,x}\,\left(\frac{2\,x^2+1}{1-x^2}\right)=\frac{4\,x}{1-x^2}+\frac{2\,x\,\left(2\,x^2+1\right)}{\left(1-x^2\right)^2}$$

```
>plot2d(["a(x)","da(x)"],r=5,color=[green,red]): //grafik fungsi dan turuna
```



Fungsi 2

$$b(x)=x\sqrt{2x+1}$$

```
>function b(x)&=x*sqrt(2*x+1)
```

```
x sqrt(2 x + 1)
```

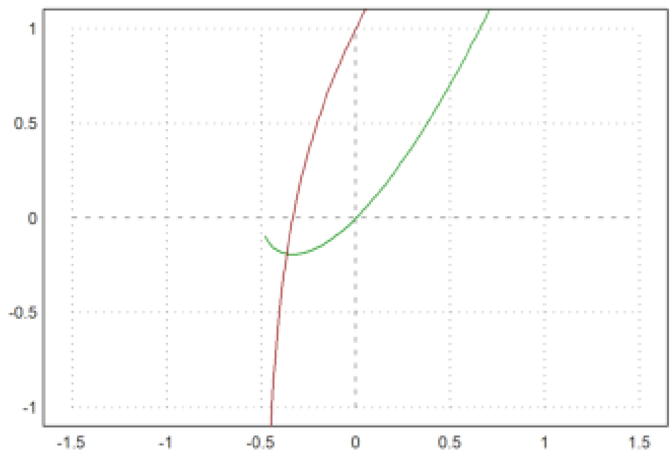
```
>function db(x) &= limit((b(x+h)-b(x))/h,h,0); $db(x) // db(x) = b'(x) meng
```

$$\frac{3\,x+1}{\sqrt{2\,x+1}}$$

```
>$&showev('diff(b(x),x))//turunan menggunakan diff
```

$$\frac{d}{d\,x}\,\left(x\,\sqrt{2\,x+1}\right)=\sqrt{2\,x+1}+\frac{x}{\sqrt{2\,x+1}}$$

```
>plot2d(["b(x)","db(x)"],r=1,color=[green,red]): //grafik fungsi dan turuna
```



Fungsi 3

$$c(x)=\frac{x+1}{x^2-5x+6}$$

```
>function c(x) &=(x+1)/(x^2-5*x+6)
```

```
                x + 1
             ------------
              2
             x  - 5 x + 6
```

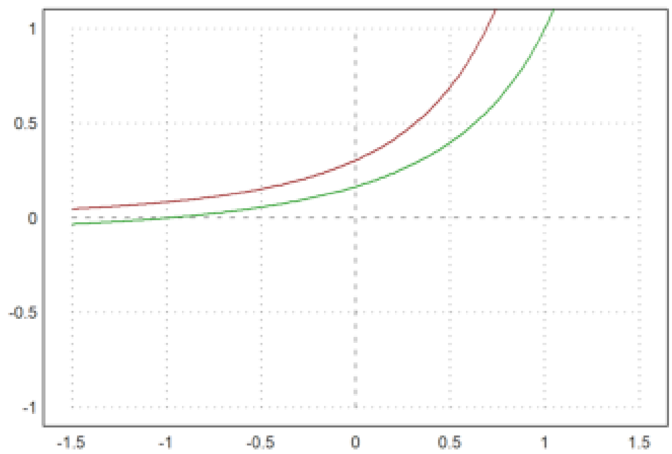
```
>function dc(x) &= limit((c(x+h)-c(x))/h,h,0); $dc(x) // dc(x) = c'(x) meng
```

$$\frac{-x^2-2\,x+11}{x^4-10\,x^3+37\,x^2-60\,x+36}$$

```
>$&showev('diff(c(x),x))//turunan menggunakan diff
```

$$\frac{d}{d\,x}\,\left(\frac{x+1}{x^2-5\,x+6}\right)=\frac{1}{x^2-5\,x+6}-\frac{\left(x+1\right)\,\left(2\,x-5\right)}{\left(x^2-5\,x+6\right)^2}$$

```
>plot2d(["c(x)","dc(x)"],r=1,color=[green,red]): //grafik fungsi dan turuna
```



Fungsi 4

$$f(x)=\sin(x^2)$$

```
>function d(x)&=sin(x^2)
```

```
                                      2
                               sin(x )
```

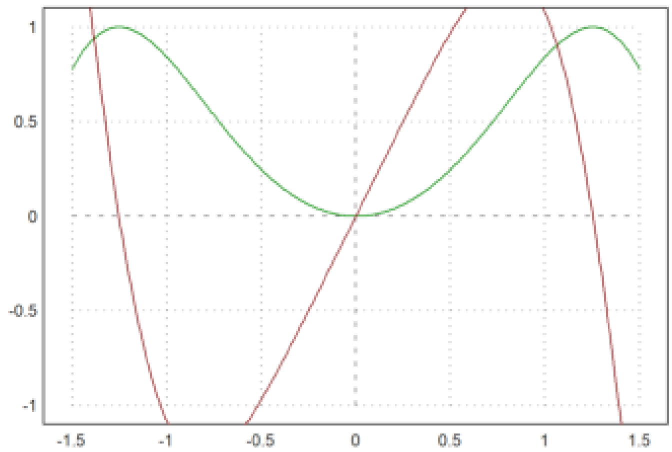
```
>function dd(x) &= limit((d(x+h)-d(x))/h,h,0); $dd(x) // dd(x) = d'(x) meng
```

$$2\,x\,\cos x^2$$

```
>$&showev('diff(d(x),x))//turunan menggunakan diff
```

$$\frac{d}{d\,x}\,\sin x^2=2\,x\,\cos x^2$$

```
>plot2d(["d(x)","dd(x)"],r=1,color=[green,red]): //grafik fungsi dan turuna
```



Fungsi 5

$$e(x)=\cos^2(\frac{x^2+2}{x^2-2})$$

```
>function e(x)&=(cos((x^2+2)/(x^2-2)))^2
```

```
                     2
                  2 x  + 2
               cos (------)
                     2
                    x  - 2
```

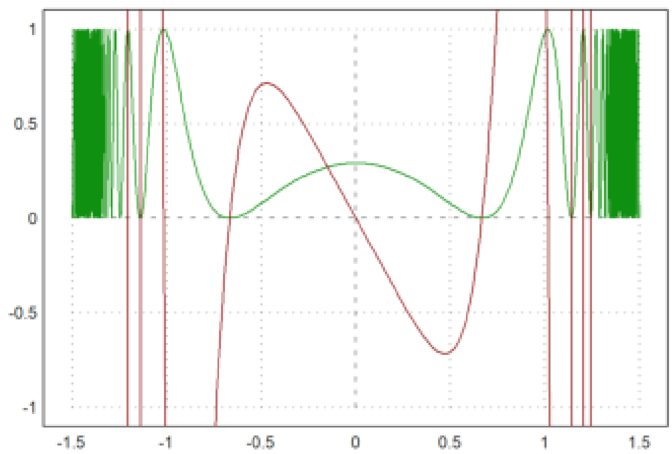
```
>function de(x) &= limit((e(x+h)-e(x))/h,h,0); $de(x) // de(x) = e'(x) meng
```

$$\frac{16\,x\,\cos\left(\frac{x^2+2}{x^2-2}\right)\,\sin\left(\frac{x^2+2}{x^2-2}\right)}{x^4-4\,x^2+4}$$

```
>$&showev('diff(e(x),x))//turunan menggunakan diff
```

$$\frac{d}{d\,x}\,\cos^2\left(\frac{x^2+2}{x^2-2}\right)=-2\,\left(\frac{2\,x}{x^2-2}-\frac{2\,x\,\left(x^2+2\right)}{\left(x^2-2\right)^2}\right)\,\cos\left(\frac{x^2+2}{x^2-2}\right)\,\sin\left(\frac{x^2+2}{x^2-2}\right)$$

```
>plot2d(["e(x)","de(x)"],r=1,color=[green,red]): //grafik fungsi dan turuna
```



Integral

EMT dapat digunakan untuk menghitung integral, baik integral tak tentu maupun integral tentu. Untuk integral tak tentu (simbolik) sudah tentu EMT menggunakan Maxima, sedangkan untuk perhitungan integral tentu EMT sudah menyediakan beberapa fungsi yang mengimplementasikan algoritma kuadratur (perhitungan integral tentu menggunakan metode numerik).

Pada notebook ini akan ditunjukkan perhitungan integral tentu dengan menggunakan Teorema Dasar Kalkulus:

$$\int_a^b f(x)\,dx=F(b)-F(a),\quad \text{dengan } F'(x)=f(x).$$

Fungsi untuk menentukan integral adalah integrate. Fungsi ini dapat digunakan untuk menentukan, baik integral tentu maupun tak tentu (jika fungsinya memiliki antiderivatif). Untuk perhitungan integral tentu fungsi integrate menggunakan metode numerik (kecuali fungsinya tidak integrabel, kita tidak akan menggunakan metode ini).

```
>$showev('integrate(x^n,x))
```

```
Answering "Is n equal to -1?" with "no"
```

$$\int x^n\,dx=\frac{x^{n+1}}{n+1}$$

```
>$showev('integrate(1/(1+x),x))
```

$$\int \frac{1}{x+1}\,dx=\log\left(x+1\right)$$

```
>$showev('integrate(1/(1+x^2),x))
```

$$\int \frac{1}{x^2+1}\,dx=\arctan x$$

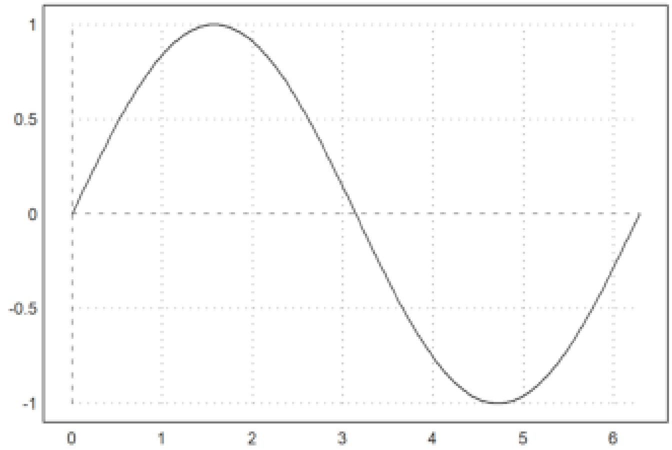
```
>$showev('integrate(1/sqrt(1-x^2),x))
```

$$\int \frac{1}{\sqrt{1-x^2}}\,dx=\arcsin x$$

```
>$showev('integrate(sin(x),x,0,pi))
```

$$\int_0^{\pi}\sin x\,dx=2$$

```
>plot2d("sin(x)",0,2*pi):
```




```
>$showev('integrate(sin(x),x,a,b))
```

$$\int_{a}^{b}{\sin x\;dx}=\cos a-\cos b$$

```
>$showev('integrate(x^n,x,a,b))
```

```
Answering "Is n positive, negative or zero?" with "positive"
```

$$\int_{a}^{b}{x^{n}\;dx}=\frac{b^{n+1}}{n+1}-\frac{a^{n+1}}{n+1}$$

```
>$showev('integrate(x^2*sqrt(2*x+1),x))
```

$$\int {x^2\,\sqrt{2\,x+1}}{\;dx}=\frac{\left(2\,x+1\right)^{\frac{7}{2}}}{28}-\frac{\left(2\,x+1\right)^{\frac{5}{2}}}{10}+\frac{\left(2\,x+1\right)^{\frac{3}{2}}}{12}$$

```
>$showev('integrate(x^2*sqrt(2*x+1),x,0,2))
```

$$\int_{0}^{2}{x^2\,\sqrt{2\,x+1}\;dx}=\frac{2\,5^{\frac{5}{2}}}{21}-\frac{2}{105}$$

```
>$ratsimp(%)
```

$$\int_{0}^{2}{x^2\,\sqrt{2\,x+1}\;dx}=\frac{2\,5^{\frac{7}{2}}-2}{105}$$

```
>$showev('integrate((sin(sqrt(x)+a)*E^sqrt(x))/sqrt(x),x,0,pi^2))
```

$$\int_{0}^{\pi^2}{\frac{\sin \left(\sqrt{x}+a\right)\,e^{\sqrt{x}}}{\sqrt{x}}\;dx}=\left(-e^{\pi}-1\right)\,\sin a+\left(e^{\pi}+1\right)\,\cos a$$

```
>$factor(%)
```

$$\int_{0}^{\pi^2}{\frac{\sin \left(\sqrt{x}+a\right)\,e^{\sqrt{x}}}{\sqrt{x}}\;dx}=\left(-e^{\pi}-1\right)\,\left(\sin a-\cos a\right)$$

```
>function map f(x) &= E^(-x^2)
```

```
                                    2
                                 - x
                                E
```

```
>$showev('integrate(f(x),x))
```

$$\int {e^ {- x^2 }}{\;dx}=\frac{\sqrt{\pi}\,\mathrm{erf}\left(x\right)}{2}$$

Fungsi f tidak memiliki antiturunan, integralnya masih memuat integral lain.

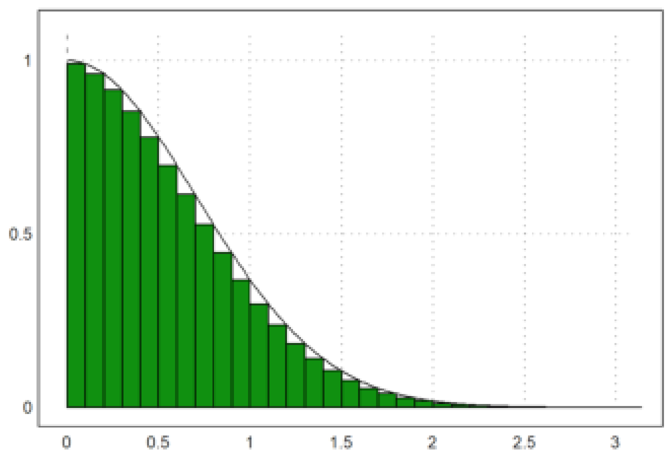
$$erf(x) = \int \frac{e^{-x^2}}{\sqrt{\pi}}\, dx.$$

Kita tidak dapat menggunakan teorema Dasar kalkulus untuk menghitung integral tentu fungsi tersebut jika semua batasnya berhingga. Dalam hal ini dapat digunakan metode numerik (rumus kuadratur).

Misalkan kita akan menghitung:

$$\int_0^{\pi} e^{-x^2}\, dx$$

```
>x=0:0.1:pi-0.1; plot2d(x,f(x+0.1),>bar); plot2d("f(x)",0,pi,>add):
```



Integral tentu

$$\int_0^{\pi} e^{-x^2}\, dx$$

dapat dihampiri dengan jumlah luas persegi-persegi panjang di bawah kurva y=f(x) tersebut. Langkah-langkahnya adalah sebagai berikut.

```
>t &= makelist(a,a,0,pi-0.1,0.1); // t sebagai list untuk menyimpan nilai-n
>fx &= makelist(f(t[i]+0.1),i,1,length(t)); // simpan nilai-nilai f(x)
>// jangan menggunakan x sebagai list, kecuali Anda pakar Maxima!
```

Hasilnya adalah:

$$\int_0^{\pi} e^{-x^2}\, dx = 0.8362196102528469$$

Jumlah tersebut diperoleh dari hasil kali lebar sub-subinterval (=0.1) dan jumlah nilai-nilai f(x) untuk x = 0.1, 0.2, 0.3, ..., 3.2.

```
>0.1*sum(f(x+0.1)) // cek langsung dengan perhitungan numerik EMT
```

```
0.836219610253
```

Untuk mendapatkan nilai integral tentu yang mendekati nilai sebenarnya, lebar sub-intervalnya dapat diperkecil lagi, sehingga daerah di bawah kurva tertutup semuanya, misalnya dapat digunakan lebar subinterval 0.001. (Silakan dicoba!)

Meskipun Maxima tidak dapat menghitung integral tentu fungsi tersebut untuk batas-batas yang berhingga, namun integral tersebut dapat dihitung secara eksak jika batas-batasnya tak hingga. Ini adalah salah satu keajaiban di dalam matematika, yang terbatas tidak dapat dihitung secara eksak, namun yang tak hingga malah dapat dihitung secara eksak.

```
>$showev('integrate(f(x),x,0,inf))
```

$$\int_{0}^{\infty }{e^ {- x^2 }\;dx}=\frac{\sqrt{\pi}}{2}$$

Tunjukkan kebenaran hasil di atas!

Berikut adalah contoh lain fungsi yang tidak memiliki antiderivatif, sehingga integral tentunya hanya dapat dihitung dengan metode numerik.

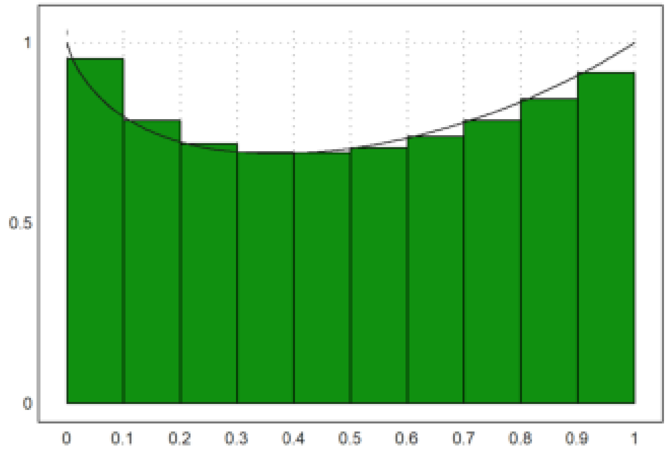
```
>function f(x) &= x^x
```

```
                                    x
                                   x
```

```
>$showev('integrate(f(x),x,0,1))
```

$$\int_{0}^{1}{x^{x}\;dx}=\int_{0}^{1}{x^{x}\;dx}$$

```
>x=0:0.1:1-0.01; plot2d(x,f(x+0.01),>bar); plot2d("f(x)",0,1,>add):
```



Maxima gagal menghitung integral tentu tersebut secara langsung menggunakan perintah integrate. Berikut kita lakukan seperti contoh sebelumnya untuk mendapat hasil atau pendekatan nilai integral tentu tersebut.

```
>t &= makelist(a,a,0,1-0.01,0.01);
>fx &= makelist(f(t[i]+0.01),i,1,length(t));
```

$$\int_0^1 x^x\, dx = 0.7834935879025506$$

Apakah hasil tersebut cukup baik? perhatikan gambarnya.

```
>function f(x) &= sin(3*x^5+7)^2
```

```
   2      5
sin  (3 x  + 7)
```

```
>integrate(f,0,1)
```

```
0.542581176074
```

```
>&showev('integrate(f(x),x,0,1))
```

```
      1                            1              pi
     /                     gamma(-) sin(14) sin(--)
     [     2    5                  5              10
     I  sin (3 x  + 7) dx = ------------------------
     ]                                   1/5
     /                                10 6
      0
    4/5                  1          4/5                   1
 - (((6    gamma_incomplete(-, 6 I) + 6    gamma_incomplete(-, - 6 I))
                         5                                5
          4/5                    1
 sin(14) + (6    I gamma_incomplete(-, 6 I)
                                 5
   4/5                  1                       pi
 - 6    I gamma_incomplete(-, - 6 I)) cos(14)) sin(--) - 60)/120
                        5                       10
```

```
>&float(%)
```

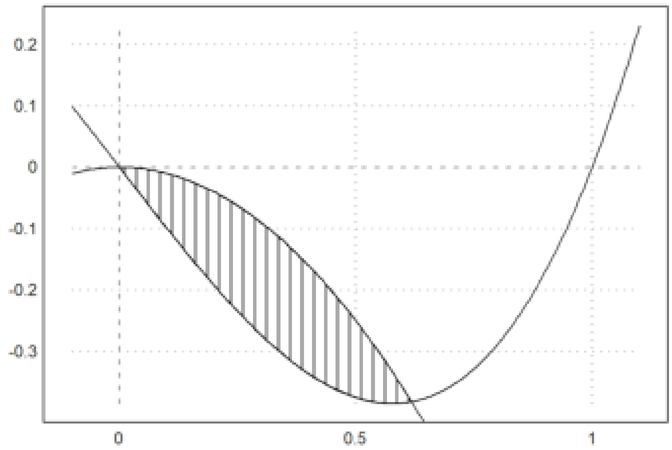
```
      1.0
     /
     [       2      5
     I    sin (3.0 x  + 7.0) dx =
     ]
     /
      0.0
0.09820784258795788 - 0.008333333333333333
 (0.3090169943749474 (0.1367372182078336
 (4.192962712629476 I gamma__incomplete(0.2, 6.0 I)
 - 4.192962712629476 I gamma__incomplete(0.2, - 6.0 I))
 + 0.9906073556948704 (4.192962712629476 gamma__incomplete(0.2, 6.0 I)
 + 4.192962712629476 gamma__incomplete(0.2, - 6.0 I))) - 60.0)
```

```
>$showev('integrate(x*exp(-x),x,0,1)) // Integral tentu (eksak)
```

$$\int_{0}^{1}{x\,e^ {- x }\;dx}=1-2\,e^ {- 1 }$$

Aplikasi Integral Tentu

```
>plot2d("x^3-x",-0.1,1.1); plot2d("-x^2",>add);  ...
>b=solve("x^3-x+x^2",0.5); x=linspace(0,b,200); xi=flipx(x); ...
>plot2d(x|xi,x^3-x|-xi^2,>filled,style="|",fillcolor=1,>add): // Plot daera
```




```
>a=solve("x^3-x+x^2",0), b=solve("x^3-x+x^2",1) // absis titik-titik potong
```

```
0
0.61803398875
```

```
>integrate("(-x^2)-(x^3-x)",a,b) // luas daerah yang diarsir
```

```
0.0758191713542
```

Hasil tersebut akan kita bandingkan dengan perhitungan secara analitik.

```
>a &= solve((-x^2)-(x^3-x),x); $a // menentukan absis titik potong kedua ku
```

$$\left[x=\frac{-\sqrt{5}-1}{2},x=\frac{\sqrt{5}-1}{2},x=0\right]$$

```
>$showev('integrate(-x^2-x^3+x,x,0,(sqrt(5)-1)/2)) // Nilai integral secara
```

$$\int_{0}^{\frac{\sqrt{5}-1}{2}}{-x^3-x^2+x\;dx}=\frac{13-5^{\frac{3}{2}}}{24}$$

```
>$float(%)
```

$$\int_{0.0}^{0.6180339887498949}-1.0\,x^3-1.0\,x^2+x\,dx=0.07581917135421037$$

Panjang Kurva

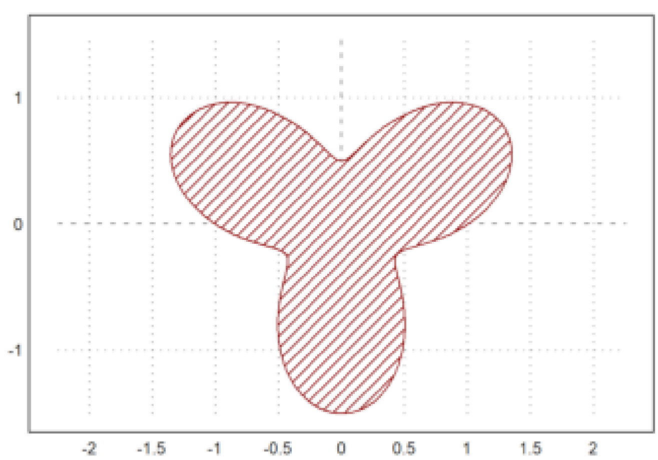
Hitunglah panjang kurva berikut ini dan luas daerah di dalam kurva tersebut.

$$\gamma(t)=(r(t)\cos(t),r(t)\sin(t))$$

dengan

$$r(t)=1+\frac{\sin(3t)}{2},\quad 0\leq t\leq 2\pi.$$

```
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t); ...
>plot2d(x,y,>filled,fillcolor=red,style="/",r=1.5): // Kita gambar kurvanya
```




```
>function r(t) &= 1+sin(3*t)/2; $'r(t)=r(t)
```

$r\,([0,0.01,0.02,0.03,0.04,0.05,0.06,0.07,0.08,0.09,0.1,0.11,0.12,0.13,0.14,0.15,0.16,0.17,0.18,0.19,0.2,$

```
>function fx(t) &= r(t)*cos(t); $'fx(t)=fx(t)
```

$$fx\left(\left[0,0.01,0.02,0.03,0.04,0.05,0.06,0.07,0.08,0.09,0.1,0.11,0.12,0.13,0.14,0.15,0.16,0.17,0.18,0.19,0.2,\right.\right.$$

```
>function fy(t) &= r(t)*sin(t); $'fy(t)=fy(t)
```

$$fy\left(\left[0,0.01,0.02,0.03,0.04,0.05,0.06,0.07,0.08,0.09,0.1,0.11,0.12,0.13,0.14,0.15,0.16,0.17,0.18,0.19,0.2,\right.\right.$$

```
>function ds(t) &= trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
... e(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); $'ds(t)=ds(t ...
                                                     ^
```

```
>$integrate(ds(x),x,0,2*pi) //panjang (keliling) kurva
```

$$\int_{0}^{2\,\pi}{ds\left(x\right)\;dx}$$

Maxima gagal melakukan perhitungan eksak integral tersebut.
Berikut kita hitung integralnya secara umerik dengan perintah EMT.

```
>integrate("ds(x)",0,2*pi)
```

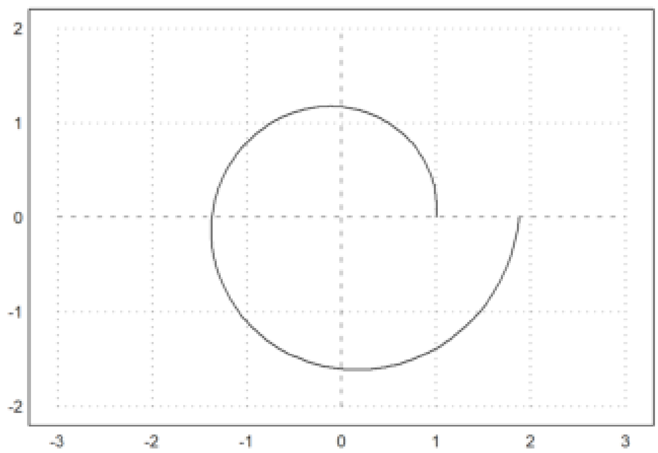
```
Function ds not found.
Try list ... to find functions!
Error in expression: ds(x)
%mapexpression1:
    return expr(x,args());
Error in map.
%evalexpression:
    if maps then return %mapexpression1(x,f$;args());
```

```
gauss:
    if maps then y=%evalexpression(f$,a+h-(h*xn)',maps;args());
adaptivegauss:
    t1=gauss(f$,c,c+h;args(),=maps);
Try "trace errors" to inspect local variables after errors.
integrate:
    return adaptivegauss(f$,a,b,eps*1000;args(),=maps);
```

Spiral Logaritmik

$$x = e^{ax} \cos x, \ y = e^{ax} \sin x.$$

```
>a=0.1; plot2d("exp(a*x)*cos(x)","exp(a*x)*sin(x)",r=2,xmin=0,xmax=2*pi):
```




```
>&kill(a) // hapus expresi a
```

done

```
>function fx(t) &= exp(a*t)*cos(t); $'fx(t)=fx(t)
```

$fx\,([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2,$

```
>function fy(t) &= exp(a*t)*sin(t); $'fy(t)=fy(t)
```

$fy\,([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2,$

```
>function df(t) &= trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
... e(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); $'df(t)=df(t ...
                                                     ^
```

```
>S &=integrate(df(t),t,0,2*%pi); $S // panjang kurva (spiral)
```

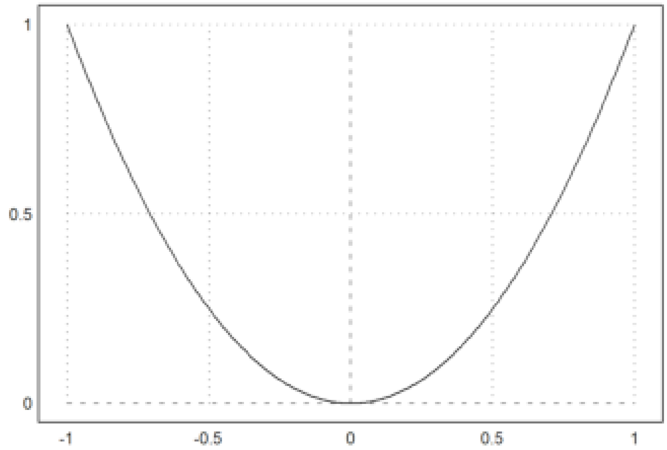
```
Maxima said:
defint: variable of integration cannot be a constant; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
S &=integrate(df(t),t,0,2*%pi); $S // panjang kurva (spiral) ...
                              ^
```

```
>S(a=0.1) // Panjang kurva untuk a=0.1
```

```
Function S not found.
Try list ... to find functions!
Error in:
S(a=0.1) // Panjang kurva untuk a=0.1 ...
        ^
```

Soal:
Tunjukkan bahwa keliling lingkaran dengan jari-jari r adalah K=2.pi.r.
Berikut adalah contoh menghitung panjang parabola.

```
>plot2d("x^2",xmin=-1,xmax=1):
```



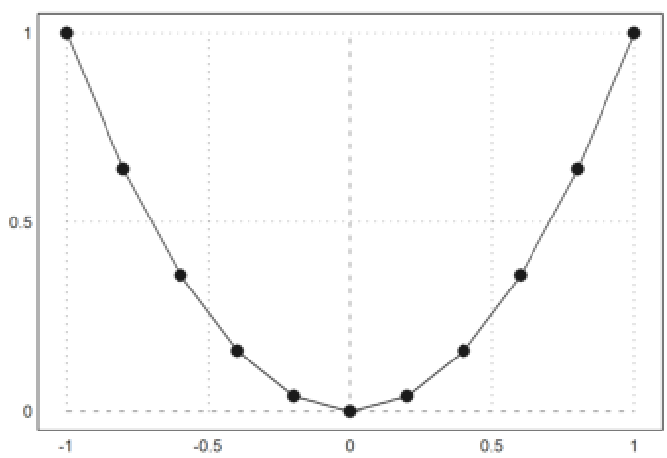

```
>$showev('integrate(sqrt(1+diff(x^2,x)^2),x,-1,1))
```

$$\int_{-1}^{1}\sqrt{4\,x^2+1}\,dx=\frac{\operatorname{asinh}2+2\sqrt{5}}{2}$$

```
>$float(%)
```

$$\int_{-1.0}^{1.0}\sqrt{4.0\,x^2+1.0}\,dx=2.957885715089195$$

```
>x=-1:0.2:1; y=x^2; plot2d(x,y);  ...
>  plot2d(x,y,points=1,style="o#",add=1):
```



Panjang tersebut dapat dihampiri dengan menggunakan jumlah panjang ruas-ruas garis yang menghubungkan titik-titik pada parabola tersebut.

```
>i=1:cols(x)-1; sum(sqrt((x[i+1]-x[i])^2+(y[i+1]-y[i])^2))
```

```
2.95191957027
```

Hasilnya mendekati panjang yang dihitung secara eksak. Untuk mendapatkan hampiran yang cukup akurat, jarak antar titik dapat diperkecil, misalnya 0.1, 0.05, 0.01, dan seterusnya. Cobalah Anda ulangi perhitungannya dengan nilai-nilai tersebut.

Koordinat Kartesius

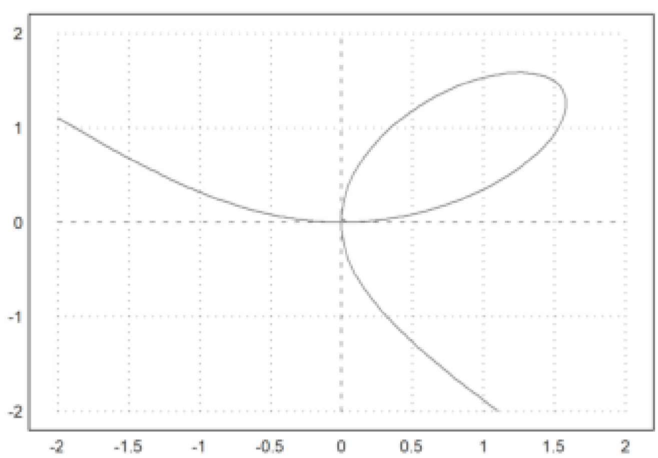
Berikut diberikan contoh perhitungan panjang kurva menggunakan koordinat Kartesius. Kita akan hitung panjang kurva dengan persamaan implisit:

$$x^3 + y^3 - 3xy = 0.$$

```
>z &= x^3+y^3-3*x*y; $z
```

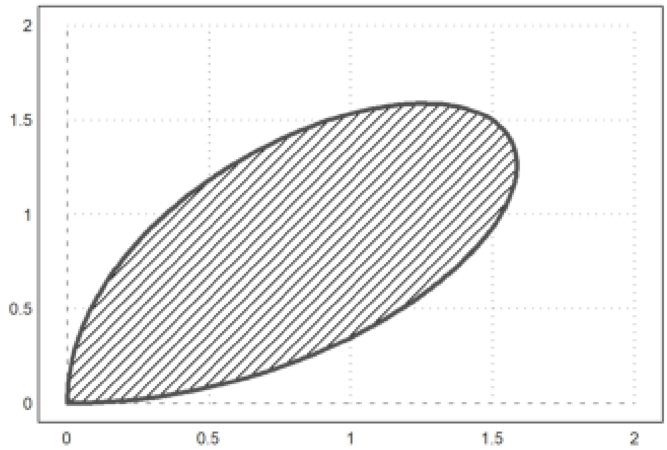
$$y^3 - 3\,x\,y + x^3$$

```
>plot2d(z,r=2,level=0,n=100):
```



Kita tertarik pada kurva di kuadran pertama.

```
>plot2d(z,a=0,b=2,c=0,d=2,level=[-10;0],n=100,contourwidth=3,style="/"):
```



Kita selesaikan persamaannya untuk x.

```
>$z with y=l*x, sol &= solve(%,x); $sol
```

$$l^3\,x^3+x^3-3\,l\,x^2$$

$$\left[x=\frac{3\,l}{l^3+1} , x=0\right]$$

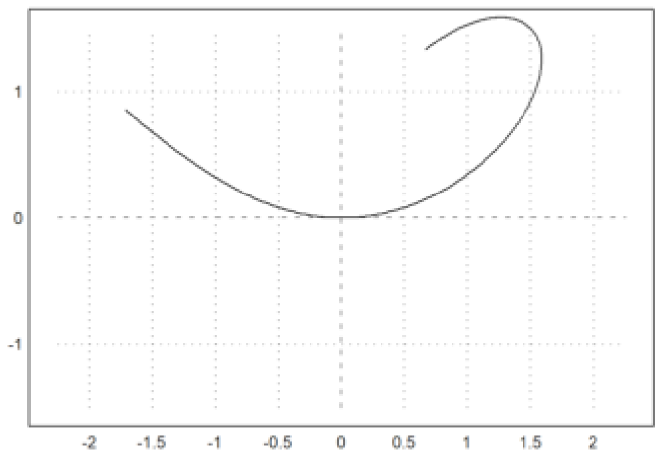
Kita gunakan solusi tersebut untuk mendefinisikan fungsi dengan Maxima.

```
>function f(l) &= rhs(sol[1]); $'f(l)=f(l)
```

$$f\left(l\right)=\frac{3\,l}{l^3+1}$$

Fungsi tersebut juga dapat digunaka untuk menggambar kurvanya. Ingat, bahwa fungsi tersebut adalah nilai x dan nilai y=l*x, yakni x=f(l) dan y=l*f(l).

```
>plot2d(&f(x),&x*f(x),xmin=-0.5,xmax=2,a=0,b=2,c=0,d=2,r=1.5):
```



Elemen panjang kurva adalah:

$$ds = \sqrt{f'(l)^2 + (lf'(l) + f(l))^2}.$$

```
>function ds(l) &= ratsimp(sqrt(diff(f(l),l)^2+diff(l*f(l),l)^2)); $'ds(l)=
```

$$ds\left(l\right) = \frac{\sqrt{9\,l^8 + 36\,l^6 - 36\,l^5 - 36\,l^3 + 36\,l^2 + 9}}{\sqrt{l^{12} + 4\,l^9 + 6\,l^6 + 4\,l^3 + 1}}$$

```
>$integrate(ds(l),l,0,1)
```

$$\int_{0}^{1}{\frac{\sqrt{9\,l^8 + 36\,l^6 - 36\,l^5 - 36\,l^3 + 36\,l^2 + 9}}{\sqrt{l^{12} + 4\,l^9 + 6\,l^6 + 4\,l^3 + 1}}\;dl}$$

Integral tersebut tidak dapat dihitung secara eksak menggunakan Maxima. Kita hitung integral etrsebut secara numerik dengan Euler. Karena kurva simetris, kita hitung untuk nilai variabel integrasi dari 0 sampai 1, kemudian hasilnya dikalikan 2.

```
>2*integrate("ds(x)",0,1)
```

```
4.91748872168
```

```
>2*romberg(&ds(x),0,1)// perintah Euler lain untuk menghitung nilai hampira
```

```
4.91748872168
```

Perhitungan di datas dapat dilakukan untuk sebarang fungsi x dan y dengan mendefinisikan fungsi EMT, misalnya kita beri nama panjangkurva. Fungsi ini selalu memanggil Maxima untuk menurunkan fungsi yang diberikan.

```
>function panjangkurva(fx,fy,a,b) ...
```

```
ds=mxm("sqrt(diff(@fx,x)^2+diff(@fy,x)^2)");
return romberg(ds,a,b);
endfunction
```

```
>panjangkurva("x","x^2",-1,1) // cek untuk menghitung panjang kurva parabol
```

```
2.95788571509
```

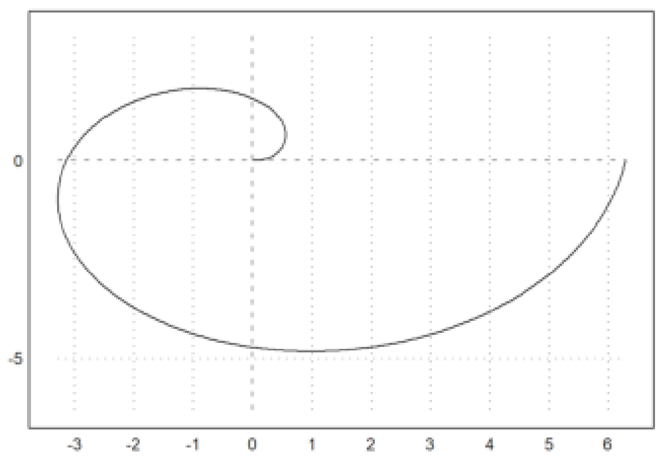
Bandingkan dengan nilai eksak di atas.

```
>2*panjangkurva(mxm("f(x)"),mxm("x*f(x)"),0,1) // cek contoh terakhir, band
```

```
4.91748872168
```

Kita hitung panjang spiral Archimides berikut ini dengan fungsi tersebut.

```
>plot2d("x*cos(x)","x*sin(x)",xmin=0,xmax=2*pi,square=1):
```



```
>panjangkurva("x*cos(x)","x*sin(x)",0,2*pi)
```

```
21.2562941482
```

Berikut kita definisikan fungsi yang sama namun dengan Maxima, untuk perhitungan eksak.

```
>&kill(ds,x,fx,fy)
```

```
done
```

```
>function ds(fx,fy) &&= sqrt(diff(fx,x)^2+diff(fy,x)^2)
```

```
                2                2
sqrt(diff  (fy, x) + diff  (fx, x))
```

```
>sol &= ds(x*cos(x),x*sin(x)); $sol // Kita gunakan untuk menghitung panjan
```

$$\sqrt{\left(\cos x-x\,\sin x\right)^2+\left(\sin x+x\,\cos x\right)^2}$$

```
>$sol | trigreduce | expand, $integrate(%,x,0,2*pi), %()
```

$$\sqrt{x^2+1}$$

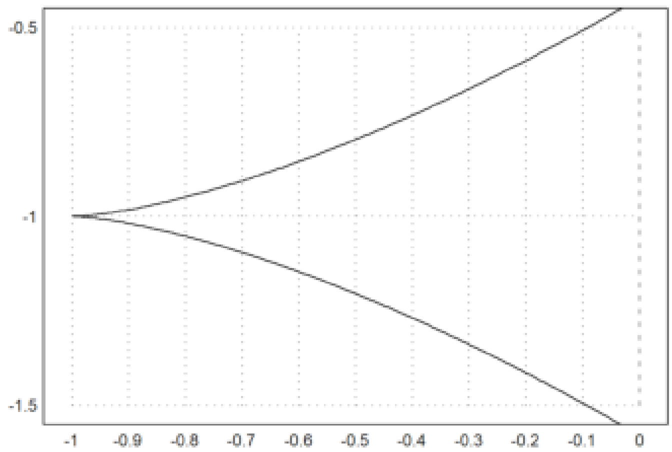
$$\frac{\operatorname{asinh}\,\left(2\,\pi\right)+2\,\pi\,\sqrt{4\,\pi^2+1}}{2}$$

```
21.2562941482
```

Hasilnya sama dengan perhitungan menggunakan fungsi EMT.

Berikut adalah contoh lain penggunaan fungsi Maxima tersebut.

```
>plot2d("3*x^2-1","3*x^3-1",xmin=-1/sqrt(3),xmax=1/sqrt(3),square=1):
```




```
>sol &= radcan(ds(3*x^2-1,3*x^3-1)); $sol
```

$$3\,x\,\sqrt{9\,x^2+4}$$

```
>$showev('integrate(sol,x,0,1/sqrt(3))), $2*float(%) // panjang kurva di at
```

$$3\int_{0}^{\frac{1}{\sqrt{3}}}{x\,\sqrt{9\,x^2+4}\;dx}=3\,\left(\frac{7^{\frac{3}{2}}}{27}-\frac{8}{27}\right)$$

$$6.0\int_{0.0}^{0.5773502691896258}{x\,\sqrt{9.0\,x^2+4.0}\;dx}=2.337835372767141$$

Sikloid

Berikut kita akan menghitung panjang kurva lintasan (sikloid) suatu titik pada lingkaran yang berputar ke kanan pada permukaan datar. Misalkan jari-jari lingkaran tersebut adalah r. Posisi titik pusat lingkaran pada saat t adalah:

$$(rt,r).$$

Misalkan posisi titik pada lingkaran tersebut mula-mula (0,0) dan posisinya pada saat t adalah:

$$(r(t-\sin(t)),r(1-\cos(t))).$$

Berikut kita plot lintasan tersebut dan beberapa posisi lingkaran ketika t=0, t=pi/2, t=r*pi.

```
>x &= r*(t-sin(t))
```

```
        [0, 1.666658333335744e-7 r, 1.333306666920222e-6 r, 
  4.499797504338432e-6 r, 1.066581336583994e-5 r, 
  2.083072932167196e-5 r, 3.599352055540239e-5 r, 
  5.71526624672386e-5 r, 8.530603082730626e-5 r, 
  1.214508019889565e-4 r, 1.665833531718508e-4 r, 
  2.216991628251896e-4 r, 2.877927110806339e-4 r, 
  3.658573803051457e-4 r, 4.568853557635201e-4 r, 
  5.618675264007778e-4 r, 6.817933857540259e-4 r, 
  8.176509330039827e-4 r, 9.704265741758145e-4 r, 
  0.001141105023499428 r, 0.001330669204938795 r, 
  0.001540100153900437 r, 0.001770376919130678 r, 
  0.002022476464811601 r, 0.002297373572865413 r, 
  0.002596040745477063 r, 0.002919448107844891 r, 
  0.003268563311168871 r, 0.003644351435886262 r, 
  0.004047774895164447 r, 0.004479793338660443 r, 0.0049413635565565 r, 
  0.005433439383882244 r, 0.005956971605131645 r, 
  0.006512907859185624 r, 0.007102192544548636 r, 
  0.007725766724910044 r, 0.008384568035038010 r, 
  0.009079530587017326 r, 0.009811584876838586 r, 0.0105816576913495 r, 
  0.01139067201557714 r, 0.01223954694042984 r, 0.01312919757078923 r, 
  0.01406053493400045 r, 0.01503446588876983 r, 0.01605189303448024 r, 
  0.01711371462093175 r, 0.01822082445851714 r, 0.01937411182884202 r, 
  0.02057446139579705 r, 0.02182275311709253 r, 0.02311986215626333 r, 
  0.02446665879515308 r, 0.02586400834688696 r, 0.02731277106934082 r, 
  0.02881380207911666 r, 0.03036795126603076 r, 0.03197606320812652 r, 
  0.0336389770872163 r, 0.03535752660496472 r, 0.03713253989951881 r, 
  0.03896483946269502 r, 0.0408552420577305 r, 0.04280455863760801 r, 
  0.04481359426396048 r, 0.04688314802656623 r, 0.04901401296344043 r, 
  0.05120697598153157 r, 0.05346281777803219 r, 0.05578231276230905 r, 
  0.05816622897846346 r, 0.06061532802852698 r, 0.0631303649963022 r, 
  0.06571208837185505 r, 0.06836123997666599 r, 0.07107855488944881 r, 
  0.07386476137264342 r, 0.07672058079958999 r, 0.07964672758239233 r, 
  0.08264390910047736 r, 0.0857128256298576 r, 0.08885417027310427 r, 
  0.09206862889003742 r, 0.09535688002914089 r, 0.0987195948597075 r, 
  0.1021574371047232 r, 0.1056710629744951 r, 0.1092611211010309 r, 
  0.1129282524731764 r, 0.1166730903725168 r, 0.1204962603100498 r, 
  0.1243983799636342 r, 0.1283800591162231 r, 0.1324418995948859 r, 
  0.1365844952106265 r, 0.140808431699002 r, 0.1451142866615502 r, 
  0.1495026295080298 r, 0.1539740213994798 r]
```

```
>y &= r*(1-cos(t))
```

```
        [0, 4.999958333473664e-5 r, 1.999933334222437e-4 r,
  4.499662510124569e-4 r, 7.998933390220841e-4 r,
  0.001249739605033717 r, 0.00179946006479581 r,
  0.002448999746720415 r, 0.003198293697380561 r,
  0.004047266988005727 r, 0.004995834721974179 r,
  0.006043902043303184 r, 0.00719136414613375 r, 0.00843810628521191 r,
  0.009784003787362772 r, 0.01122892206395776 r, 0.01277271662437307 r,
  0.01441523309043924 r, 0.01615630721187855 r, 0.01799576488272969 r,
  0.01993342215875837 r, 0.02196908527585173 r, 0.02410255066939448 r,
  0.02633360499462523 r, 0.02866202514797045 r, 0.03108757828935527 r,
  0.03361002186548678 r, 0.03622910363410947 r, 0.03894456168922911 r,
  0.04175612448730281 r, 0.04466351087439402 r, 0.04766643011428662 r,
  0.05076458191755917 r, 0.0539576564716131 r, 0.05724533447165381 r,
  0.06062728715262111 r, 0.06410317632206519 r, 0.06767265439396564 r,
  0.07133536442348987 r, 0.07509094014268702 r, 0.07893900599711501 r,
  0.08287917718339499 r, 0.08691105968769186 r, 0.09103425032511492 r,
  0.09524833678003664 r, 0.09955289764732322 r, 0.1039475024744748 r,
  0.1084317118046711 r, 0.113005077220716 r, 0.1176671413898787 r,
  0.1224174381096274 r, 0.1272554923542488 r, 0.1321808203223502 r,
  0.1371929294852391 r, 0.1422913186361759 r, 0.1474754779404944 r,
  0.152744888986584 r, 0.1580990248377314 r, 0.1635373500848132 r,
  0.1690593208998367 r, 0.1746643850903219 r, 0.1803519821545206 r,
  0.1861215433374662 r, 0.1919724916878484 r, 0.1979042421157076 r,
  0.2039162014509444 r, 0.2100077685026351 r, 0.216178334119151 r,
  0.2224272812490723 r, 0.2287539850028937 r, 0.2351578127155118 r,
  0.2416381240094921 r, 0.2481942708591053 r, 0.2548255976551299 r,
  0.2615314412704124 r, 0.2683111311261794 r, 0.2751639892590951 r,
  0.2820893303890569 r, 0.2890864619877229 r, 0.2961546843477643 r,
  0.303293290652834 9 r, 0.3105015670482534 r, 0.3177787927123868 r,
  0.3251242399287333 r, 0.3325371741586922 r, 0.3400168541150183 r,
  0.3475625318359485 r, 0.3551734527599992 r, 0.3628488558014202 r,
  0.3705879734263036 r, 0.3783900317293359 r, 0.3862542505111889 r,
  0.3941798433565377 r, 0.4021660177127022 r, 0.4102119749689023 r,
  0.418316910536117 r, 0.4264800139275439 r, 0.4347004688396462 r,
  0.4429774532337832 r, 0.451310139418413 r]
```

Berikut kita gambar sikloid untuk r=1.

```
>ex &= x-sin(x); ey &= 1-cos(x); aspect(1);
>plot2d(ex,ey,xmin=0,xmax=4pi,square=1); ...
```

```
>  plot2d("2+cos(x)","1+sin(x)",xmin=0,xmax=2pi,>add,color=blue); ...
>  plot2d([2,ex(2)],[1,ey(2)],color=red,>add); ...
>  plot2d(ex(2),ey(2),>points,>add,color=red); ...
>  plot2d("2pi+cos(x)","1+sin(x)",xmin=0,xmax=2pi,>add,color=blue); ...
>  plot2d([2pi,ex(2pi)],[1,ey(2pi)],color=red,>add);  ...
>  plot2d(ex(2pi),ey(2pi),>points,>add,color=red):
```

```
Error : [0,1.66665833335744e-7*r-sin(1.66665833335744e-7*r),1.33330666692
```

```
>ds &= radcan(sqrt(diff(ex,x)^2+diff(ey,x)^2)); $ds=trigsimp(ds) // elemen
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
ds &= radcan(sqrt(diff(ex,x)^2+diff(ey,x)^2)); $ds=trigsimp(ds ...
                                              ^
```

```
>ds &= trigsimp(ds); $ds
>$showev('integrate(ds,x,0,2*pi)) // hitung panjang sikloid satu putaran pe
```

```
Maxima said:
defint: variable of integration must be a simple or subscripted variable.
defint: found errexp1
#0: showev(f='integrate(ds,[0,1.66665833335744e-7*r,1.33330666692022e-6*r
 -- an error. To debug this try: debugmode(true);

Error in:
$showev('integrate(ds,x,0,2*pi)) // hitung panjang sikloid sat ...
                                ^
```

```
>integrate(mxm("ds"),0,2*pi) // hitung secara numerik
```

```
Illegal function result in map.
%evalexpression:
    if maps then return %mapexpression1(x,f$;args());
gauss:
    if maps then y=%evalexpression(f$,a+h-(h*xn)',maps;args());
adaptivegauss:
    t1=gauss(f$,c,c+h;args(),=maps);
Try "trace errors" to inspect local variables after errors.
integrate:
    return adaptivegauss(f$,a,b,eps*1000;args(),=maps);
```

```
>romberg(mxm("ds"),0,2*pi) // cara lain hitung secara numerik
```

```
Wrong argument!

Cannot combine a symbolic expression here.
Did you want to create a symbolic expression?
Then start with &.

Try "trace errors" to inspect local variables after errors.
romberg:
    if cols(y)==1 then return y*(b-a); endif;
Error in:
romberg(mxm("ds"),0,2*pi) // cara lain hitung secara numerik ...
                         ^
```

Perhatikan, seperti terlihat pada gambar, panjang sikloid lebih besar daripada keliling lingkarannya, yakni:

$$2\pi.$$

Kurvatur (Kelengkungan) Kurva

image: Osculating.png

Aslinya, kelengkungan kurva diferensiabel (yakni, kurva mulus yang tidak lancip) di titik P didefinisikan melalui lingkaran oskulasi (yaitu, lingkaran yang melalui titik P dan terbaik memperkirakan, paling banyak menyinggung kurva di sekitar P). Pusat dan radius kelengkungan kurva di P adalah pusat dan radius lingkaran oskulasi. Kelengkungan adalah kebalikan dari radius kelengkungan:

$$\kappa = \frac{1}{R}$$

dengan R adalah radius kelengkungan. (Setiap lingkaran memiliki kelengkungan ini pada setiap titiknya, dapat diartikan, setiap lingkaran berputar 2pi sejauh 2piR.)
Definisi ini sulit dimanipulasi dan dinyatakan ke dalam rumus untuk kurva umum. Oleh karena itu digunakan definisi lain yang ekivalen.

Definisi Kurvatur dengan Fungsi Parametrik Panjang Kurva

Setiap kurva diferensiabel dapat dinyatakan dengan persamaan parametrik terhadap panjang kurva s:

$$\gamma(s) = (x(s),\ y(s)),$$

dengan x dan y adalah fungsi riil yang diferensiabel, yang memenuhi:

$$\|\gamma'(s)\| = \sqrt{x'(s)^2 + y'(s)^2} = 1.$$

Ini berarti bahwa vektor singgung

$$\mathbf{T}(s) = (x'(s),\ y'(s))$$

memiliki norm 1 dan merupakan vektor singgung satuan.

Apabila kurvanya memiliki turunan kedua, artinya turunan kedua x dan y ada, maka T'(s) ada. Vektor ini merupakan normal kurva yang arahnya menuju pusat kurvatur, norm-nya merupakan nilai kurvatur (kelengkungan):

$$\begin{aligned}\mathbf{T}(s) &= \gamma'(s),\\ \mathbf{T}^2(s) &= 1 \text{ (konstanta)} \Rightarrow \mathbf{T}'(s)\cdot\mathbf{T}(s) = 0\\ \kappa(s) &= \|\mathbf{T}'(s)\| = \|\gamma''(s)\| = \sqrt{x''(s)^2 + y''(s)^2}.\end{aligned}$$

Nilai

$$R(s) = \frac{1}{\kappa(s)}$$

disebut jari-jari (radius) kelengkungan kurva.

Bilangan riil

$$k(s) = \pm\kappa(s)$$

disebut nilai kelengkungan bertanda.

Contoh:
Akan ditentukan kurvatur lingkaran

$$x = r\cos t,\ y = r\sin t.$$

```
>fx &= r*cos(t); fy &=r*sin(t);
>&assume(t>0,r>0); s &=integrate(sqrt(diff(fx,t)^2+diff(fy,t)^2),t,0,t); s
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
... =integrate(sqrt(diff(fx,t)^2+diff(fy,t)^2),t,0,t); s // elemen ...
                                                     ^
```

```
>&kill(s); fx &= r*cos(s/r); fy &=r*sin(s/r); // definisi ulang persamaan p
>k &= trigsimp(sqrt(diff(fx,s,2)^2+diff(fy,s,2)^2)); $k // nilai kurvatur l
```

Untuk representasi parametrik umum, misalkan

$$x = x(t),\ y = y(t)$$

merupakan persamaan parametrik untuk kurva bidang yang terdiferensialkan dua kali. Kurvatur untuk kurva tersebut didefinisikan sebagai

$$\begin{aligned}\kappa &= \frac{d\phi}{ds} = \frac{\frac{d\phi}{dt}}{\frac{ds}{dt}} \quad (\phi \text{ adalah sudut kemiringan garis singgung dan } s \text{ adalah panjang kurva}) \\ &= \frac{\frac{d\phi}{dt}}{\sqrt{(\frac{dx}{dt})^2 + (\frac{dy}{dt})^2}} = \frac{\frac{d\phi}{dt}}{\sqrt{x'(t)^2 + y'(t)^2}}.\end{aligned}$$

Selanjutnya, pembilang pada persamaan di atas dapat dicari sebagai berikut.

$$\sec^2\phi\frac{d\phi}{dt}=\frac{d}{dt}\left(\tan\phi\right)=\frac{d}{dt}\left(\frac{dy}{dx}\right)=\frac{d}{dt}\left(\frac{dy/dt}{dx/dt}\right)=\frac{d}{dt}\left(\frac{y'(t)}{x'(t)}\right)=\frac{x'(t)y''(t)-x''(t)y'(t)}{x'(t)^2}.$$

$$\begin{aligned}\frac{d\phi}{dt}&=\frac{1}{\sec^2\phi}\frac{x'(t)y''(t)-x''(t)y'(t)}{x'(t)^2}\\&=\frac{1}{1+\tan^2\phi}\frac{x'(t)y''(t)-x''(t)y'(t)}{x'(t)^2}\\&=\frac{1}{1+\left(\frac{y'(t)}{x'(t)}\right)^2}\frac{x'(t)y''(t)-x''(t)y'(t)}{x'(t)^2}\\&=\frac{x'(t)y''(t)-x''(t)y'(t)}{x'(t)^2+y'(t)^2}.\end{aligned}$$

Jadi, rumus kurvatur untuk kurva parametrik

$$x=x(t),\ y=y(t)$$

adalah

$$\kappa(t)=\frac{x'(t)y''(t)-x''(t)y'(t)}{\left(x'(t)^2+y'(t)^2\right)^{3/2}}.$$

Jika kurvanya dinyatakan dengan persamaan parametrik pada koordinat kutub

$$x=r(\theta)\cos\theta,\ y=r(\theta)\sin\theta,$$

maka rumus kurvaturnya adalah

$$\kappa(\theta)=\frac{r(\theta)^2+2r'(\theta)^2-r(\theta)r''(\theta)}{\left(r'(\theta)^2+r'(\theta)^2\right)^{3/2}}.$$

(Silakan Anda turunkan rumus tersebut!)

Contoh:

Lingkaran dengan pusat (0,0) dan jari-jari r dapat dinyatakan dengan persamaan parametrik

$$x=r\cos t,\ y=r\sin t.$$

Nilai kelengkungan lingkaran tersebut adalah

$$\kappa(t)=\frac{x'(t)y''(t)-x''(t)y'(t)}{\left(x'(t)^2+y'(t)^2\right)^{3/2}}=\frac{r^2}{r^3}=\frac{1}{r}.$$

Hasil cocok dengan definisi kurvatur suatu kelengkungan.

Kurva

$$y=f(x)$$

dapat dinyatakan ke dalam persamaan parametrik

$$x = t,\ y = f(t),\ \text{ dengan } x'(t) = 1,\ x''(t) = 0,$$

sehingga kurvaturnya adalah

$$\kappa(t) = \frac{y''(t)}{(1 + y'(t)^2)^{3/2}}.$$

Contoh:
Akan ditentukan kurvatur parabola

$$y = ax^2 + bx + c.$$

```
>function f(x) &= a*x^2+b*x+c; $y=f(x)
```

```
>function k(x) &= (diff(f(x),x,2))/(1+diff(f(x),x)^2)^(3/2); $'k(x)=k(x) //
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
... (x) &= (diff(f(x),x,2))/(1+diff(f(x),x)^2)^(3/2); $'k(x)=k(x)  ...
                                                     ^
```

```
>function f(x) &= x^2+x+1; $y=f(x) // akan kita plot kelengkungan parabola
```

```
>function k(x) &= (diff(f(x),x,2))/(1+diff(f(x),x)^2)^(3/2); $'k(x)=k(x) //
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);
```

```
Error in:
... (x) &= (diff(f(x),x,2))/(1+diff(f(x),x)^2)^(3/2); $'k(x)=k(x)  ...
                                                        ^
```

Berikut kita gambar parabola tersebut beserta kurva kelengkungan, kurva jari-jari kelengkungan dan salah satu lingkaran oskulasi di titik puncak parabola. Perhatikan, puncak parabola dan jari-jari lingkaran oskulasi di puncak parabola adalah

$$(-1/2, 3/4),\ 1/k(2) = 1/2,$$

sehingga pusat lingkaran oskulasi adalah (-1/2, 5/4).

```
>plot2d(["f(x)", "k(x)"],-2,1, color=[blue,red]); plot2d("1/k(x)",-1.5,1,co
>plot2d("-1/2+1/k(-1/2)*cos(x)","5/4+1/k(-1/2)*sin(x)",xmin=0,xmax=2pi,>add
```

```
Error : f(x) does not produce a real or column vector

Error generated by error() command

%ploteval:
    error(f$|" does not produce a real or column vector");
adaptiveevalone:
    s=%ploteval(g$,t;args());
Try "trace errors" to inspect local variables after errors.
plot2d:
    dw/n,dw/n^2,dw/n,auto;args());
```

Untuk kurva yang dinyatakan dengan fungsi implisit

$$F(x, y) = 0$$

dengan turunan-turunan parsial

$$F_x = \frac{\partial F}{\partial x},\ F_y = \frac{\partial F}{\partial y},\ F_{xy} = \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial x}\right),\ F_{xx} = \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial x}\right),\ F_{yy} = \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial y}\right),$$

berlaku

$$F_x dx + F_y dy = 0 \text{ atau } \frac{dy}{dx} = -\frac{F_x}{F_y},$$

sehingga kurvaturnya adalah

$$\kappa = \frac{F_y^2 F_{xx} - 2F_x F_y F_{xy} + F_x^2 F_{yy}}{\left(F_x^2 + F_y^2\right)^{3/2}}.$$

(Silakan Anda turunkan sendiri!)

Contoh 1:
Parabola

$$y = ax^2 + bx + c$$

dapat dinyatakan ke dalam persamaan implisit

$$ax^2 + bx + c - y = 0.$$

```
>function F(x,y) &=a*x^2+b*x+c-y; $F(x,y)
```

```
>Fx &= diff(F(x,y),x), Fxx &=diff(F(x,y),x,2), Fy &=diff(F(x,y),y), Fxy &=d
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
Fx &= diff(F(x,y),x), Fxx &=diff(F(x,y),x,2), Fy &=diff(F(x,y) ...
                    ^
```

```
>function k(x) &= (Fy^2*Fxx-2*Fx*Fy*Fxy+Fx^2*Fyy)/(Fx^2+Fy^2)^(3/2); $'k(x)
```

$k\left(\left[0, 1.666658333335744\times 10^{-7}\,r, 1.333306666692022\times 10^{-6}\,r, 4.499797504338432\times 10^{-6}\,r, 1.066581336583\right.\right.$

Hasilnya sama dengan sebelumnya yang menggunakan persamaan parabola biasa.

Latihan

- Bukalah buku Kalkulus.
- Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi).
- Untuk setiap fungsi, tentukan anti turunannya (jika ada), hitunglah integral tentu dengan batas-batas yang menarik (Anda tentukan sendiri), seperti contoh-contoh tersebut.

- Lakukan hal yang sama untuk fungsi-fungsi yang tidak dapat diintegralkan (cari sedikitnya 3 fungsi).
- Gambar grafik fungsi dan daerah integrasinya pada sumbu koordinat yang sama.
- Gunakan integral tentu untuk mencari luas daerah yang dibatasi oleh dua kurva yang berpotongan di dua titik. (Cari dan gambar kedua kurva dan arsir (warnai) daerah yang dibatasi oleh keduanya.)
- Gunakan integral tentu untuk menghitung volume benda putar kurva y= f(x) yang diputar mengelilingi sumbu x dari x=a sampai x=b, yakni

$$V = \int_a^b \pi(f(x)^2 \, dx.$$

(Pilih fungsinya dan gambar kurva dan benda putar yang dihasilkan. Anda dapat mencari contoh-contoh bagaimana cara menggambar benda hasil perputaran suatu kurva.)
- Gunakan integral tentu untuk menghitung panjang kurva y=f(x) dari x=a sampai x=b dengan menggunakan rumus:

$$S = \int_a^b \sqrt{1 + (f'(x))^2} \, dx.$$

(Pilih fungsi dan gambar kurvanya.)
- Apabila fungsi dinyatakan dalam koordinat kutub x=f(r,t), y=g(r,t), r=h(t), x=a bersesuaian dengan t=t0 dan x=b bersesuian dengan t=t1, maka rumus di atas akan menjadi:

$$S = \int_{t_0}^{t_1} \sqrt{x'(t)^2 + y'(t)^2} \, dt.$$

- Pilih beberapa kurva menarik (selain lingkaran dan parabola) dari buku kalkulus. Nyatakan setiap kurva tersebut dalam bentuk:

```
a. koordinat Kartesius (persamaan y=f(x))
b. koordinat kutub ( r=r(theta))
c. persamaan parametrik x=x(t), y=y(t)
d. persamaan implit F(x,y)=0
```

- Tentukan kurvatur masing-masing kurva dengan menggunakan keempat representasi tersebut (hasilnya harus sama).
- Gambarlah kurva asli, kurva kurvatur, kurva jari-jari lingkaran oskulasi, dan salah satu lingkaran oskulasinya.

JAWAB

Fungsi 1

$$a(x) = \sin 2x$$

```
>function a(x) &= sin(2*x)//mendefinisikan fungsi a
```

```
        [0, sin(3.333316666714881e-7 r), sin(2.66661333384044e-6 r),
  sin(8.999595008676864e-6 r), sin(2.133162673167988e-5 r),
  sin(4.166145864334392e-5 r), sin(7.198704111080478e-5 r),
  sin(1.143053249344772e-4 r), sin(1.706120616546125e-4 r),
  sin(2.42901603977913e-4 r), sin(3.331667063437016e-4 r),
  sin(4.433983256503793e-4 r), sin(5.755854221612677e-4 r),
  sin(7.317147606102914e-4 r), sin(9.137707115270399e-4 r),
  sin(0.001123735052801556 r), sin(0.001363586771508052 r),
  sin(0.001635301866007965 r), sin(0.001940853148351629 r),
  sin(0.002282210046998856 r), sin(0.002661338409877589 r),
  sin(0.003080200307800873 r), sin(0.003540753838261357 r),
  sin(0.004044952929623202 r), sin(0.004594747145730826 r),
  sin(0.005192081490954126 r), sin(0.005838896215689782 r),
  sin(0.006537126622337741 r), sin(0.007288702871772523 r),
  sin(0.008095549790328893 r), sin(0.008959586677320885 r),
  sin(0.009882727113112999 r), sin(0.01086687876776449 r),
  sin(0.01191394321026329 r), sin(0.01302581571837125 r),
  sin(0.01420438508909727 r), sin(0.01545153344982009 r),
  sin(0.01676913607007602 r), sin(0.01815906117403465 r),
  sin(0.01962316975367717 r), sin(0.021163315382699 r),
  sin(0.02278134403115428 r), sin(0.02447909388085967 r),
  sin(0.02625839514157846 r), sin(0.02812106986800089 r),
  sin(0.03006893177753966 r), sin(0.03210378606896047 r),
  sin(0.03422742924186351 r), sin(0.03644164891703428 r),
  sin(0.03874822365768404 r), sin(0.0411489227915941 r),
  sin(0.04364550623418506 r), sin(0.04623972431252665 r),
  sin(0.04893331759030617 r), sin(0.05172801669377391 r),
  sin(0.05462554213868165 r), sin(0.05762760415823331 r),
  sin(0.06073590253206151 r), sin(0.06395212641625303 r),
  sin(0.06727795417443261 r), sin(0.07071505320992943 r),
  sin(0.07426507979903763 r), sin(0.07792967892539004 r),
  sin(0.081710484115461 r), sin(0.08560911727521603 r),
  sin(0.08962718852792095 r), sin(0.09376629605313247 r),
  sin(0.09802802592688087 r), sin(0.1024139519630631 r),
  sin(0.1069256355560644 r), sin(0.1115646255246181 r),
  sin(0.1163324579569269 r), sin(0.121230656057054 r),
  sin(0.126260729992604 4 r), sin(0.1314241767437101 r),
```

```
sin(0.136722479953332 r), sin(0.1421571097788976 r),
sin(0.1477295227452868 r), sin(0.15344116159918 r),
sin(0.1592934551647847 r), sin(0.1652878182009547 r),
sin(0.1714256512597152 r), sin(0.1777083405462085 r),
sin(0.1841372577800748 r), sin(0.1907137600582818 r),
sin(0.197439189719415 r), sin(0.204314874209446 5 r),
sin(0.2113421259489903 r), sin(0.2185222422020618 r),
sin(0.2258565049463528 r), sin(0.2333461807450337 r),
sin(0.2409925206200996 r), sin(0.2487967599272685 r),
sin(0.2567601182324462 r), sin(0.2648837991897719 r),
sin(0.2731689904212531 r), sin(0.2816168633980041 r),
sin(0.2902285733231005 r), sin(0.2990052590160597 r),
sin(0.3079480427989596 r)]
```

```
>function ga(x) &= integrate(a(x),x); $showev('integrate(a(x),x))
```

```
Maxima output too long!
Error in:
function ga(x) &= integrate(a(x),x); $showev('integrate(a(x),x ...
                                    ^
```

```
>function gan(x)&=integrate(a(x),x,-pi,pi); $showev('integrate(a(x),x,-pi,p
```

```
Maxima said:
defint: variable of integration must be a simple or subscripted variable.
defint: found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
function gan(x)&=integrate(a(x),x,-pi,pi); $showev('integrate( ...
                                          ^
```

```
>plot2d(["a","ga","gan"],color=[red,blue,green]):
```

```
Error : a does not produce a real or column vector

Error generated by error() command

%ploteval:
    error(f$|" does not produce a real or column vector");
```

```
adaptiveevalone:
    s=%ploteval(g$,t;args());
Try "trace errors" to inspect local variables after errors.
plot2d:
    dw/n,dw/n^2,dw/n,auto;args());
```

Fungsi 2

$$b(x) = 2x + \pi$$

```
>function b(x) &= 2*x+pi
```

```
        [pi, 3.333316666714881e-7 r + pi, 2.666613333840044e-6 r + pi,
8.999595008676864e-6 r + pi, 2.133162673167988e-5 r + pi,
4.166145864334392e-5 r + pi, 7.198704111080478e-5 r + pi,
1.143053249344772e-4 r + pi, 1.706120616546125e-4 r + pi,
2.42901603977913e-4 r + pi, 3.331667063437016e-4 r + pi,
4.433983256503793e-4 r + pi, 5.755854221612677e-4 r + pi,
7.317147606102914e-4 r + pi, 9.137707115270399e-4 r + pi,
0.001123735052801556 r + pi, 0.001363586771508052 r + pi,
0.001635301866007965 r + pi, 0.001940853148351629 r + pi,
0.002282210046998856 r + pi, 0.002661338409877589 r + pi,
0.003080200307800873 r + pi, 0.003540753838261357 r + pi,
0.004044952929623202 r + pi, 0.004594747145730826 r + pi,
0.005192081490954126 r + pi, 0.005838896215689782 r + pi,
0.006537126622337741 r + pi, 0.007288702871772523 r + pi,
0.008095549790328893 r + pi, 0.008959586677320885 r + pi,
0.009882727113112999 r + pi, 0.01086687876776449 r + pi,
0.01191394321026329 r + pi, 0.01302581571837125 r + pi,
0.01420438508909727 r + pi, 0.01545153344982009 r + pi,
0.01676913607007602 r + pi, 0.01815906117403465 r + pi,
0.01962316975367717 r + pi, 0.02116331538269 r + pi,
0.02278134403115428 r + pi, 0.02447909388085967 r + pi,
0.02625839514157846 r + pi, 0.02812106986800089 r + pi,
0.03006893177753966 r + pi, 0.03210378606896047 r + pi,
0.03422742924186351 r + pi, 0.03644164891703428 r + pi,
0.03874822365768404 r + pi, 0.0411489227915941 r + pi,
0.04364550623418506 r + pi, 0.04623972431252665 r + pi,
0.04893331759030617 r + pi, 0.05172801669377391 r + pi,
0.05462554213868165 r + pi, 0.05762760415823331 r + pi,
0.06073590253206151 r + pi, 0.06395212641625303 r + pi,
0.06727795417443261 r + pi, 0.07071505320992943 r + pi,
0.07426507979903763 r + pi, 0.07792967892539004 r + pi,
```

```
0.081710484115461 r + pi, 0.0856091172752160 3 r + pi,
0.08962718852792095 r + pi, 0.09376629605313247 r + pi,
0.09802802592688087 r + pi, 0.1024139519630631 r + pi,
0.1069256355560644 r + pi, 0.1115646255246181 r + pi,
0.1163324579569269 r + pi, 0.121230656057054 r + pi,
0.1262607299926044 r + pi, 0.1314241767437101 r + pi,
0.136722479953332 r + pi, 0.1421571097788976 r + pi,
0.1477295227452868 r + pi, 0.15344116159918 r + pi,
0.1592934551647847 r + pi, 0.1652878182009547 r + pi,
0.1714256512597152 r + pi, 0.1777083405462085 r + pi,
0.1841372577800748 r + pi, 0.1907137600582818 r + pi,
0.197439189719415 r + pi, 0.2043148742094465 r + pi,
0.2113421259489903 r + pi, 0.2185222422020618 r + pi,
0.225856504946352 8 r + pi, 0.2333461807450337 r + pi,
0.2409925206200996 r + pi, 0.2487967599272685 r + pi,
0.2567601182324462 r + pi, 0.2648837991897719 r + pi,
0.2731689904212531 r + pi, 0.2816168633980041 r + pi,
0.2902285733231005 r + pi, 0.2990052590160597 r + pi,
0.3079480427989596 r + pi]
```

```
>function gb(x) &= integrate(b(x),x);$showev('integrate(b(x),x))
```

```
Maxima output too long!
Error in:
function gb(x) &= integrate(b(x),x);$showev('integrate(b(x),x) ...
                                        ^
```

```
>function gbn(x)&=integrate(b(x),x,1,2); $showev('integrate(b(x),x,1,2))
```

```
Maxima said:
defint: variable of integration must be a simple or subscripted variable.
defint: found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
function gbn(x)&=integrate(b(x),x,1,2); $showev('integrate(b(x ...
                                           ^
```

```
>plot2d(["b","gb","gbn"],color=[red,blue,green]):
```

```
Error : b does not produce a real or column vector

Error generated by error() command

%ploteval:
    error(f$|" does not produce a real or column vector");
adaptiveevalone:
    s=%ploteval(g$,t;args());
Try "trace errors" to inspect local variables after errors.
plot2d:
    dw/n,dw/n^2,dw/n,auto;args());
```

Fungsi 3

$$c(x) = x^2 + 2$$

```
>function c(x) &= x^2+2;
>function gc(x) &= integrate(c(x),x); $showev('integrate(c(x),x))
```

```
Maxima output too long!
Error in:
function gc(x) &= integrate(c(x),x); $showev('integrate(c(x),x ...
                                                ^
```

```
>function gcn(x)&=integrate(c(x),x,0,1); $showev('integrate(c(x),x,0,1))
```

```
Maxima said:
defint: variable of integration must be a simple or subscripted variable.
defint: found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
function gcn(x)&=integrate(c(x),x,0,1); $showev('integrate(c(x ...
                                           ^
```

```
>plot2d(["c","gc","gcn"],color=[red,blue,green]):
```

```
Error : c does not produce a real or column vector

Error generated by error() command
```

```
%ploteval:
    error(f$|" does not produce a real or column vector");
adaptiveevalone:
    s=%ploteval(g$,t;args());
Try "trace errors" to inspect local variables after errors.
plot2d:
    dw/n,dw/n^2,dw/n,auto;args());
```

Fungsi 4

$$g(x) = \sqrt{x^3} + 4$$

```
>function d(x) &= sqrt(x^3)+4
```

```
                                           3/2
        [4, 6.804087143572822e-11 r    + 4, 
                         3/2                              3/2
1.53955453048757e-9 r    + 4, 9.545297216017534e-9 r    + 4, 
                          3/2                              3/2
3.483300723327515e-8 r    + 4, 9.507289389712205e-8 r    + 4, 
                          3/2                              3/2
2.159416876226504e-7 r    + 4, 4.320705844220854e-7 r    + 4, 
                          3/2                              3/2
7.878972831748384e-7 r    + 4, 1.338445156559387e-6 r    + 4, 
                          3/2                              3/2
2.150044257308648e-6 r    + 4, 3.301004221415825e-6 r    + 4, 
                          3/2                              3/2
4.882246306738607e-6 r    + 4, 6.997899973513604e-6 r    + 4, 
                          3/2                              3/2
9.765868164967462e-6 r    + 4, 1.331836456218631e-5 r    + 4, 
                          3/2                              3/2
1.780242544149966e-5 r    + 4, 2.338039827842887e-5 r    + 4, 
                          3/2                              3/2
3.023040887125708e-5 r    + 4, 3.854680846778374e-5 r    + 4, 
                          3/2                              3/2
4.854060214924248e-5 r    + 4, 6.043985954082078e-5 r    + 4, 
                          3/2                              3/2
7.449010876799885e-5 r    + 4, 9.095471445439255e-5 r    + 4, 
                          3/2                              3/2
1.101152404541769e-4 r    + 4, 1.322717979262492e-4 r    + 4, 
                          3/2                              3/2
1.577433792847805e-4 r    + 4, 1.868681784991794e-4 r    + 4, 
```

```
                  3/2                            3/2
2.200038981638455e-4 r    + 4, 2.575280437128157e-4 r    + 4,
                  3/2                            3/2
2.998382051152764e-4 r    + 4, 3.473523263539594e-4 r    + 4,
                  3/2                            3/2
4.005089629589743e-4 r    + 4, 4.597675278435574e-4 r    + 4,
                  3/2                            3/2
5.256085256657785e-4 r    + 4, 5.985337759200305e-4 r    + 4,
                  3/2                            3/2
6.790666249447842e-4 r    + 4, 7.677521470171642e-4 r    + 4,
                  3/2                           3/2
8.651573346915501e-4 r    + 4, 9.71871278526663e-4 r    + 4,
                  3/2                            3/2
0.001088505336335157 r    + 4, 0.001215693292079795 r    + 4,
                 3/2                            3/2
0.00135409150453165 r    + 4, 0.00150437904579836 4 r    + 4,
                  3/2                            3/2
0.001667257829823287 r    + 4, 0.001843452730950423 r    + 4,
                  3/2                            3/2
0.002033711692644811 r    + 4, 0.002238805826452798 r    + 4,
                  3/2                            3/2
0.002459529501282533 r    + 4, 0.002696700423081472 r    + 4,
                  3/2                            3/2
0.002951159704983676 r    + 4, 0.003223771927997423 r    + 4,
                  3/2                            3/2
0.003515425192300452 r    + 4, 0.003827031159208176 r    + 4,
                  3/2                            3/2
0.004159525083878177 r    + 4, 0.004513865838812381 r    + 4,
                  3/2                            3/2
0.004891035928217165 r    + 4, 0.005292041493279701 r    + 4,
                  3/2                            3/2
0.005717912308419052 r    + 4, 0.006169701768567838 r    + 4,
                  3/2                            3/2
0.006648486867541619 r    + 4, 0.007155368167550851 r    + 4,
                  3/2                            3/2
0.007691469759911013 r    + 4, 0.008257939217005645 r    + 4,
                  3/2                            3/2
0.008855947535557414 r    + 4, 0.009486689071261337 r    + 4,
                3/2                            3/2
0.0101513814648359 r    + 4, 0.0108512655595462 8 r    + 4,
                 3/2                            3/2
0.01158760531025531 r    + 4, 0.01236168768405805 r    + 4,
                 3/2                            3/2
0.01317482255255527 r    + 4, 0.01402834257582326 r    + 4,
                 3/2                            3/2
0.01492360307813616 r    + 4, 0.01586198191549924 r    + 4,
```

```
                     3/2                           3/2
0.01684487933505093 r    + 4, 0.01787371782639267 r    + 4,
                     3/2                           3/2
0.01894994196490681 r    + 4, 0.02007501824712262 r    + 4,
                     3/2                           3/2
0.02125043491819205 r    + 4, 0.02247770179153755 r    + 4,
                     3/2                           3/2
0.02375835006073511 r    + 4, 0.02509393210369652 r    + 4,
                     3/2                          3/2
0.02648602127921605 r    + 4, 0.0279362117159475 r    + 4,
                     3/2                           3/2
0.02944611809387885 r    + 4, 0.03101737541837212 r    + 4,
                     3/2                           3/2
0.03265163878683829 r    + 4, 0.03435058314811649 r    + 4,
                     3/2                           3/2
0.03611590305462957 r    + 4, 0.03794931240738771 r    + 4,
                     3/2                           3/2
0.03985254419391367 r    + 4, 0.04182735021916435 r    + 4,
                     3/2                           3/2
0.04387550082952382 r    + 4, 0.04599878462994463 r    + 4,
                     3/2                           3/2
0.04819900819431567 r    + 4, 0.05047799576913461 r    + 4,
                    3/2                           3/2
0.0528375889705655 r    + 4, 0.05527964647496281 r    + 4,
                     3/2                           3/2
0.05780604370294355 r    + 4, 0.06041867249709122 r    + 4]
```

```
>function gd(x) &= integrate(d(x),x); $showev('integrate(d(x),x))
```

```
Maxima output too long!
Error in:
function gd(x) &= integrate(d(x),x); $showev('integrate(d(x),x ...
                                    ^
```

```
>function gdn(x)&=integrate(d(x),x,0,4); $showev('integrate(d(x),x,0,4))
```

```
Maxima said:
defint: variable of integration must be a simple or subscripted variable.
defint: found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
```

```
function gdn(x)&=integrate(d(x),x,0,4); $showev('integrate(d(x ...
                                         ^
```

```
>plot2d(["d","gd","gdn"],color=[red,blue,green]):
```

```
Error : d does not produce a real or column vector

Error generated by error() command

%ploteval:
    error(f$|" does not produce a real or column vector");
adaptiveevalone:
    s=%ploteval(g$,t;args());
Try "trace errors" to inspect local variables after errors.
plot2d:
    dw/n,dw/n^2,dw/n,auto;args());
```

Fungsi 5

$$e(x) = \cos 2x - 3$$

```
>function e(x) &= cos(2*x)-3
```

```
        [- 2, cos(3.333316666714881e-7 r) - 3,
cos(2.66661333384044e-6 r) - 3, cos(8.999595008676864e-6 r) - 3,
cos(2.133162673167988e-5 r) - 3, cos(4.166145864334392e-5 r) - 3,
cos(7.198704111080478e-5 r) - 3, cos(1.143053249344772e-4 r) - 3,
cos(1.706120616546125e-4 r) - 3, cos(2.42901603977913e-4 r) - 3,
cos(3.331667063437016e-4 r) - 3, cos(4.433983256503793e-4 r) - 3,
cos(5.755854221612677e-4 r) - 3, cos(7.317147606102914e-4 r) - 3,
cos(9.137707115270399e-4 r) - 3, cos(0.001123735052801556 r) - 3,
cos(0.001363586771508052 r) - 3, cos(0.001635301866007965 r) - 3,
cos(0.001940853148351629 r) - 3, cos(0.002282210046998856 r) - 3,
cos(0.002661338409877589 r) - 3, cos(0.003080200307800873 r) - 3,
cos(0.003540753838261357 r) - 3, cos(0.004044952929623202 r) - 3,
cos(0.004594747145730826 r) - 3, cos(0.005192081490954126 r) - 3,
cos(0.005838896215689782 r) - 3, cos(0.006537126622337741 r) - 3,
cos(0.007288702871772523 r) - 3, cos(0.008095549790328893 r) - 3,
cos(0.008959586677320885 r) - 3, cos(0.009882727113112999 r) - 3,
cos(0.01086687876776449 r) - 3, cos(0.01191394321026329 r) - 3,
cos(0.01302581571837125 r) - 3, cos(0.01420438508909727 r) - 3,
cos(0.0154515334498209 r) - 3, cos(0.01676913607007602 r) - 3,
```

```
cos(0.01815906117403465 r) - 3, cos(0.01962316975367717 r) - 3,
cos(0.021163315382699 r) - 3, cos(0.02278134403115428 r) - 3,
cos(0.02447909388085967 r) - 3, cos(0.02625839514157846 r) - 3,
cos(0.02812106986800089 r) - 3, cos(0.03006893177753966 r) - 3,
cos(0.03210378606896047 r) - 3, cos(0.03422742924186351 r) - 3,
cos(0.03644164891703428 r) - 3, cos(0.03874822365768404 r) - 3,
cos(0.0411489227915941 r) - 3, cos(0.04364550623418506 r) - 3,
cos(0.04623972431252665 r) - 3, cos(0.04893331759030617 r) - 3,
cos(0.05172801669377391 r) - 3, cos(0.05462554213868165 r) - 3,
cos(0.05762760415823331 r) - 3, cos(0.06073590253206151 r) - 3,
cos(0.06395212641625303 r) - 3, cos(0.06727795417443261 r) - 3,
cos(0.07071505320992943 r) - 3, cos(0.07426507979903763 r) - 3,
cos(0.07792967892539004 r) - 3, cos(0.081710484115461 r) - 3,
cos(0.08560911727521603 r) - 3, cos(0.08962718852792095 r) - 3,
cos(0.09376629605313247 r) - 3, cos(0.09802802592688087 r) - 3,
cos(0.1024139519630631 r) - 3, cos(0.1069256355560644 r) - 3,
cos(0.1115646255246181 r) - 3, cos(0.1163324579569269 r) - 3,
cos(0.121230656057054 r) - 3, cos(0.1262607299926044 r) - 3,
cos(0.1314241767437101 r) - 3, cos(0.136722479953332 r) - 3,
cos(0.1421571097788976 r) - 3, cos(0.1477295227452868 r) - 3,
cos(0.15344116159918 r) - 3, cos(0.1592934551647847 r) - 3,
cos(0.1652878182009547 r) - 3, cos(0.1714256512597152 r) - 3,
cos(0.1777083405462085 r) - 3, cos(0.1841372577800748 r) - 3,
cos(0.1907137600582818 r) - 3, cos(0.197439189719415 r) - 3,
cos(0.2043148742094465 r) - 3, cos(0.2113421259489903 r) - 3,
cos(0.2185222422020618 r) - 3, cos(0.2258565049463528 r) - 3,
cos(0.2333461807450337 r) - 3, cos(0.2409925206200996 r) - 3,
cos(0.2487967599272685 r) - 3, cos(0.256760118232462 r) - 3,
cos(0.2648837991897719 r) - 3, cos(0.2731689904212531 r) - 3,
cos(0.2816168633980041 r) - 3, cos(0.2902285733231005 r) - 3,
cos(0.2990052590160597 r) - 3, cos(0.3079480427989596 r) - 3]
```

```
>function ge(x) &= integrate(e(x),x); $showev('integrate(e(x),x))
```

```
Maxima output too long!
Error in:
function ge(x) &= integrate(e(x),x); $showev('integrate(e(x),x ...
                                                ^
```

```
>function gen(x)&=integrate(e(x),x,0,pi); $showev('integrate(e(x),x,0,pi))
```

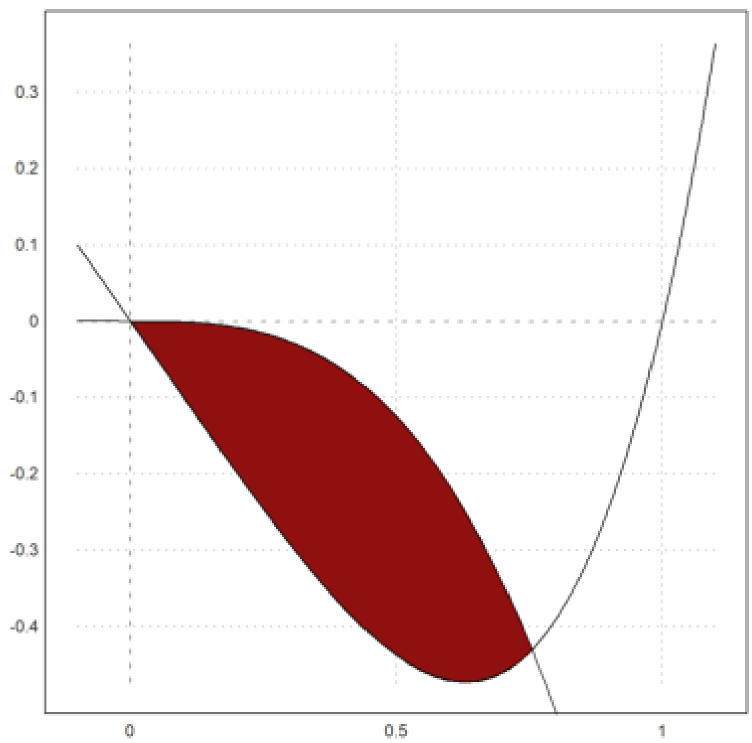
```
Maxima said:
defint: variable of integration must be a simple or subscripted variable.
defint: found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
function gen(x)&=integrate(e(x),x,0,pi); $showev('integrate(e( ...
                                         ^
```

```
>plot2d(["e","ge","gen"],color=[red,blue,green]):
```

```
Error : e does not produce a real or column vector

Error generated by error() command

%ploteval:
    error(f$|" does not produce a real or column vector");
adaptiveevalone:
    s=%ploteval(g$,t;args());
Try "trace errors" to inspect local variables after errors.
plot2d:
    dw/n,dw/n^2,dw/n,auto;args());
```

Akan digunakan integral tentu untuk mencari luas daerah yang dibatasi oleh dua kurva yang berpotongan di dua titik.

```
>plot2d("x^4-x",-0.1,1.1); plot2d("-x^3",>add);  ...
>b=solve("x^4-x+x^3",0.5); x=linspace(0,b,200); xi=flipx(x); ...
>plot2d(x|xi,x^4-x|-xi^3,>filled,fillcolor=2,>add): // Plot daerah antara 2
```



Akan digunakan integral tentu untuk menghitung volume benda putar kurva

$$y = f(x) = \sqrt{x}$$

yang diputar mengelilingi sumbu x dari x=0 sampai x=4.

```
>function f(x) &= sqrt(x)
```

```
        [0, 4.082472698448135e-4 sqrt(r),
  0.001154688991425925 sqrt(r), 0.002121272614337542 sqrt(r),
  0.003265855686621798 sqrt(r), 0.004564069381776745 sqrt(r),
  0.005999460021985511 sqrt(r), 0.007559937993610702 sqrt(r),
  0.009236126397321891 sqrt(r), 0.01102047194946553 sqrt(r),
  0.0129067173662341 7 sqrt(r), 0.01488956556871924 sqrt(r),
  0.01696445434078661 sqrt(r), 0.01912739868108431 sqrt(r),
  0.02137487674265094 sqrt(r), 0.02370374498683231 sqrt(r),
  0.02611117358055792 sqrt(r), 0.02859459622033476 sqrt(r),
  0.03115167048772528 sqrt(r), 0.0337802460544536 sqrt(r),
  0.0364783388456601 sqrt(r), 0.03924410979880212 sqrt(r),
```

```
0.04207584721821628 sqrt(r), 0.0449719519791125 sqrt(r),
0.04793092501574962 sqrt(r), 0.05095135665982863 sqrt(r),
0.05403191749183894 sqrt(r), 0.05717135044031119 sqrt(r),
0.06036846391855819 sqrt(r), 0.06362212583028365 sqrt(r),
0.06693125830776261 sqrt(r), 0.07029483307154587 sqrt(r),
0.07371186732054916 sqrt(r), 0.07718142007718985 sqrt(r),
0.08070258892492621 sqrt(r), 0.08427450708576488 sqrt(r),
0.08789634079363055 sqrt(r), 0.09156728692627084 sqrt(r),
0.09528657086398548 sqrt(r), 0.09905344454807509 sqrt(r),
0.1028671847157756 sqrt(r), 0.1067270912916544 sqrt(r),
0.1106324859181508 sqrt(r), 0.1145827106102366 sqrt(r),
0.1185771265210979 sqrt(r), 0.1226151128073935 sqrt(r),
0.1266960655840592 sqrt(r), 0.130819396959823 sqrt(r),
0.1349845341456462 sqrt(r), 0.1391909186292052 sqrt(r),
0.1434380054092954 sqrt(r), 0.1477252622847309 sqrt(r),
0.1520521691928903 sqrt(r), 0.1564182175935817 sqrt(r),
0.1608229098943523 sqrt(r), 0.1652657589137593 sqrt(r),
0.1697462873794789 sqrt(r), 0.1742640274584251 sqrt(r),
0.1788185203163434 sqrt(r), 0.1834093157045637 sqrt(r),
0.1880359715718371 sqrt(r), 0.1926980536993532 sqrt(r),
0.1973951353572195 sqrt(r), 0.2021267969808321 sqrt(r),
0.2068926258657084 sqrt(r), 0.2116922158794708 sqrt(r),
0.2165251671897894 sqrt(r), 0.2213910860071842 sqrt(r),
0.2262895843416828 sqrt(r), 0.231220279772414 sqrt(r),
0.2361827952292653 sqrt(r), 0.2411767587858819 sqrt(r),
0.2462018034631895 sqrt(r), 0.2512575670428698 sqrt(r),
0.2563436918901166 sqrt(r), 0.2614598247851206 sqrt(r),
0.2666056167627547 sqrt(r), 0.271780722959969 sqrt(r),
0.2769848024704424 sqrt(r), 0.282217518206068 sqrt(r),
0.2874785367648816 sqrt(r), 0.292767528305066 sqrt(r),
0.2980841664246933 sqrt(r), 0.303428128046886 sqrt(r),
0.3087990933101017 sqrt(r), 0.3141967454632646 sqrt(r),
0.3196207707654858 sqrt(r), 0.3250708583901287 sqrt(r),
0.3305467003329952 sqrt(r), 0.3360479913244184 sqrt(r),
0.3415744287450641 sqrt(r), 0.3471257125452533 sqrt(r),
0.3527015451676307 sqrt(r), 0.3583016314730134 sqrt(r),
0.3639256786692661 sqrt(r), 0.3695733962430556 sqrt(r),
0.3752444958943462 sqrt(r), 0.3809386914735103 sqrt(r),
0.3866556989209261 sqrt(r), 0.392395236208952 7 sqrt(r)]
```

```
>plot2d("f",0,4):
```

```
Error : f does not produce a real or column vector
```

```
Error generated by error() command

%ploteval:
    error(f$|" does not produce a real or column vector");
adaptiveevalone:
    s=%ploteval(g$,t;args());
Try "trace errors" to inspect local variables after errors.
plot2d:
    dw/n,dw/n^2,dw/n,auto;args());
```

```
>function gf(x) &=pi*(f(x))^2; $'gf(x)=gf(x)
```

$$gf\left(\left[0, 1.666658333335744 \times 10^{-7}\, r, 1.333306666692022 \times 10^{-6}\, r, 4.499797504338432 \times 10^{-6}\, r, 1.06658133658\right.\right.$$

```
>$integrate(gf(x),x,0,4)//volume benda putar
```

```
Maxima said:
defint: variable of integration must be a simple or subscripted variable.
defint: found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
$integrate(gf(x),x,0,4)//volume benda putar  ...
                         ^
```

Akan digunakan integral tentu untuk menghitung panjang kurva

$$y = f(x) = x^4 - 2x^3 + 2$$

dari x=1 sampai x=2

```
>function f(x) &= x^4-2*x^3+2
```

```
                             4                           3
        [2, 7.71589506333222e-28 r  - 9.259120371466594e-21 r  + 2,
                    4                          3
```

```
3.16024099764332e-24 r  - 4.740456304689606e-18 r  + 2,
                      4                           3
4.099886953132549e-22 r  - 1.822253978842242e-16 r  + 2,
                       4                           3
1.294124084842248e-20 r  - 2.426676785826797e-15 r  + 2,
                     4                           3
1.8828594509022e-19 r  - 1.807771030794685e-14 r  + 2,
                       4                           3
1.678407106553627e-18 r  - 9.326162490663663e-14 r  + 2,
                       4                          3
1.066954421676005e-17 r  - 3.73369979845685e-13 r  + 2,
                       4                           3
5.295645941937888e-17 r  - 1.241564257668583e-12 r  + 2,
                      4                           3
2.17571270549331e-16 r  - 3.582870874234567e-12 r  + 2,
                       4                           3
7.700632522459392e-16 r  - 9.245380616771788e-12 r  + 2,
                       4                          3
2.415773498052583e-15 r  - 2.17932577396102e-11 r  + 2,
                       4                           3
6.859921745022883e-15 r  - 4.767265799932555e-11 r  + 2,
                      4                           3
1.79162569057795e-14 r  - 9.794120807860339e-11 r  + 2,
                       4                           3
4.357415285322094e-14 r  - 1.907443620310499e-10 r  + 2,
                       4                        3
9.966340703890378e-14 r  - 3.547576692226e-10 r  + 2,
                     4                           3
2.1607829029214e-13 r  - 6.338527032003085e-10 r  + 2,
                      4                          3
4.46963178314049e-13 r  - 1.093286047315919e-9 r  + 2,
                      4                          3
8.86851128500447e-13 r  - 1.827755241046757e-9 r  + 2,
                       4                          3
1.695518251365634e-12 r  - 2.971712886104008e-9 r  + 2,
                       4                          3
3.135309549847579e-12 r  - 4.712380114022087e-9 r  + 2,
                       4                          3
5.625949856680611e-12 r  - 7.305953242628289e-9 r  + 2,
                       4                         3
9.823425498495825e-12 r  - 1.10975526085366e-8 r  + 2,
                       4                          3
1.673146256382637e-11 r  - 1.654552016296017e-8 r  + 2,
                       4                         3
2.785649582389407e-11 r  - 2.42507323605624e-8 r  + 2,
                       4                          3
```

```
4.541988373104388e-11 r  - 3.499165705328501e-8 r  + 2,
                       4                          3
7.264455050990641e-11 r  - 4.976594741636421e-8 r  + 2,
                       4                          3
1.141373029972576e-10 r  - 6.983943227120236e-8 r  + 2,
                       4                          3
1.763928603150368e-10 r  - 9.680343041457541e-8 r  + 2,
                       4                          3
2.684512373637712e-10 r  - 1.326413865970998e-7 r  + 2,
                      4                          3
4.02746633161521e-10 r  - 1.798058984935011e-7 r  + 2,
                       4                          3
5.961934928601376e-10 r  - 2.41307277247015e-7 r  + 2,
                       4                          3
8.715640444261754e-10 r  - 3.208148588209461e-7 r  + 2,
                      4                          3
1.259221469948161e-9 r  - 4.227723593187526e-7 r  + 2,
                      4                          3
1.799284075611245e-9 r  - 5.52528644505107e-7 r  + 2,
                      4                          3
2.544308497548467e-9 r  - 7.164853618341787e-7 r  + 2,
                      4                          3
3.562594252598255e-9 r  - 9.222629622278006e-7 r  + 2,
                      4                          3
4.942227948428493e-9 r  - 1.178886718498931e-6 r  + 2,
                     4                          3
6.79600334672729e-9 r  - 1.496994427541174e-6 r  + 2,
                     4                          3
9.26737337138014e-9 r  - 1.889067564050101e-6 r  + 2,
                      4                          3
1.253761222072381e-8 r  - 2.369687734460225e-6 r  + 2,
                     4                          3
1.68343901335078e-8 r  - 2.95582036081562e-6 r  + 2,
                      4                          3
2.244199023074389e-8 r  - 3.667127605289572e-6 r  + 2,
                      4                          3
2.971342637269594e-8 r  - 4.526312626874392e-6 r  + 2,
                      4                          3
3.908475329784208e-8 r  - 5.559497342214114e-6 r  + 2,
                      4                          3
5.109189561793014e-8 r  - 6.796635942497141e-6 r  + 2,
                      4                          3
6.639036070214325e-8 r  - 8.271966497600447e-6 r  + 2,
                      4                          3
8.577824226808761e-8 r  - 1.002450305711799e-5 r  + 2,
                      4                          3
```

```
1.102229667839573e-7 r  - 1.209857073535821e-5 r  + 2,
                      4                          3
1.408922837523204e-7 r  - 1.454438634369558e-5 r  + 2,
                      4                          3
1.791900537698002e-7 r  - 1.741868720863868e-5 r  + 2,
                      4                          3
2.267974451174936e-7 r  - 2.078541088748844e-5 r  + 2,
                      4                          3
2.857202107126794e-7 r  - 2.471642856532134e-5 r  + 2,
                     4                          3
3.58342782721357e-7 r  - 2.929233498710054e-5 r  + 2,
                      4                          3
4.474899921700293e-7 r  - 3.460329784682351e-5 r  + 2,
                      4                          3
5.564972956786061e-7 r  - 4.074996962159482e-5 r  + 2,
                      4                          3
6.892904711369381e-7 r  - 4.784446490222228e-5 r  + 2,
                      4                          3
8.504758289340514e-7 r  - 5.60114063331881e-5 r  + 2,
                      4                          3
1.045442075388071e-6 r  - 6.538904233354016e-5 r  + 2,
                      4                          3
1.280475060475582e-6 r  - 7.613043982613823e-5 r  + 2,
                     4                          3
1.56288674298023e-6 r  - 8.840475525574673e-5 r  + 2,
                      4                          3
1.901159812919329e-6 r  - 1.023985872264001e-4 r  + 2,
                      4                          3
2.305109523710632e-6 r  - 1.183174141352512e-4 r  + 2,
                      4                          3
2.786064405144342e-6 r  - 1.363871202235196e-4 r  + 2,
                      4                          3
3.357067652958322e-6 r  - 1.568556135050909e-4 r  + 2,
                      4                          3
4.033101121796338e-6 r  - 1.799945390695785e-4 r  + 2,
                      4                          3
4.831333985674849e-6 r  - 2.061010912892276e-4 r  + 2,
                     4                          3
5.77139827389003e-6 r  - 2.354999284875906e-4 r  + 2,
                      4                          3
6.875693640660019e-6 r  - 2.685451936525142e-4 r  + 2,
                      4                          3
8.169723883794696e-6 r  - 3.056226447963851e-4 r  + 2,
                      4                          3
9.682467891383049e-6 r  - 3.471518985826378e-4 r  + 2,
                      4                          3
```

$1.144678786594879e-5\ r^{4} - 3.93588790849311e-4\ r^{3} + 2,$

$1.349987785278008e-5\ r^{4} - 4.454278576675103e-4\ r^{3} + 2,$

$1.588375578322904e-5\ r^{4} - 5.032049405752499e-4\ r^{3} + 2,$

$1.864580243470444e-5\ r^{4} - 5.674999196248515e-4\ r^{3} + 2,$

$2.183935090685177e-5\ r^{4} - 6.389395778750145e-4\ r^{3} + 2,$

$2.552433041801159e-5\ r^{4} - 7.182006009466724e-4\ r^{3} + 2,$

$2.976796843742984e-5\ r^{4} - 8.060127152446122e-4\ r^{3} + 2,$

$3.464555538681935e-5\ r^{4} - 9.031619684246317e-4\ r^{3} + 2,$

$4.024127636966925e-5\ r^{4} - 0.00101049415565858\ r^{3} + 2,$

$4.664911461807966e-5\ r^{4} - 0.001128918395216864\ r^{3} + 2,$

$5.397383158476088e-5\ r^{4} - 0.001259410856849862\ r^{3} + 2,$

$6.23320288520571e-5\ r^{4} - 0.001403018646406171\ r^{3} + 2,$

$7.185329728028343e-5\ r^{4} - 0.001560863850076484\ r^{3} + 2,$

$8.268145907415057e-5\ r^{4} - 0.001734147741597319\ r^{3} + 2,$

$9.497590870839159e-5\ r^{4} - 0.00192415515568847\ r^{3} + 2,$

$1.089130589217236e-4\ r^{4} - 0.002132259030932325\ r^{3} + 2,$

$1.246878982617065e-4\ r^{4} - 0.002359925125231328\ r^{3} + 2,$

$1.425156669417069e-4\ r^{4} - 0.002608716906902802\ r^{3} + 2,$

$1.626336580547126e-4\ r^{4} - 0.002880300624387022\ r^{3} + 2,$

$1.853031514769292e-4\ r^{4} - 0.003176450557455684\ r^{3} + 2,$

$2.108114880865876e-4\ r^{4} - 0.003499054452713256\ r^{3} + 2,$

$2.394742922198537e-4\ r^{4} - 0.003850119146083091\ r^{3} + 2,$

$2.716378505858421e-4\ r^{4} - 0.00423177637486406\ r^{3} + 2,$

```
3.076816561660808e-4 r  - 0.004646288781831418 r  + 2,
                      4                          3
3.480211259299347e-4 r  - 0.005096056113737543 r  + 2,
                      4                         3
3.931105015061231e-4 r  - 0.00558362161644485 r  + 2,
                      4                          3
4.434459422610224e-4 r  - 0.006111678628793737 r  + 2,
                      4                          3
4.995688205465132e-4 r  - 0.006683077377173238 r  + 2,
                      4                          3
5.620692291932171e-4 r  - 0.007300831972621533 r  + 2]
```

```
>plot2d("f",1,2):
```

```
Error : f does not produce a real or column vector

Error generated by error() command

 %ploteval:
    error(f$|" does not produce a real or column vector");
adaptiveevalone:
    s=%ploteval(g$,t;args());
Try "trace errors" to inspect local variables after errors.
plot2d:
    dw/n,dw/n^2,dw/n,auto;args());
```

```
>function df(x) &= diff(f(x),x); $showev('diff(f(x),x))
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
function df(x) &= diff(f(x),x); $showev('diff(f(x),x)) ...
                               ^
```

```
>function gf(x) &=(1+(df(x))^2)^(1/2); $'gf(x)=gf(x)
```

```
>$integrate(gf(x),x,1,2)//panjang kurva
```

```
Maxima said:
defint: variable of integration must be a simple or subscripted variable.
defint: found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
$integrate(gf(x),x,1,2)//panjang kurva  ...
                       ^
```

Maxima gagal melakukan perhitungan eksak integral tersebut.
Berikut akan dihitung integralnya secara numerik dengan perintah EMT.

```
>integrate("gf(x)",1,2)
```

```
Illegal function result in map.
%evalexpression:
    if maps then return %mapexpression1(x,f$;args());
gauss:
    if maps then y=%evalexpression(f$,a+h-(h*xn)',maps;args());
adaptivegauss:
    t1=gauss(f$,c,c+h;args(),=maps);
Try "trace errors" to inspect local variables after errors.
integrate:
    return adaptivegauss(f$,a,b,eps*1000;args(),=maps);
```

Jadi, panjang kurvanya adalah 2.69055132339.

Pilih beberapa kurva menarik (selain lingkaran dan parabola) dari buku kalkulus. Nyatakan setiap kurva tersebut dalam bentuk:

```
a. koordinat Kartesius (persamaan y=f(x))
b. koordinat kutub ( r=r(theta))
c. persamaan parametrik x=x(t), y=y(t)
d. persamaan implit F(x,y)=0
```

Fungsi 1

$$f(x,y) = x^3 - y^3 + 2xy$$

a. bentuk koordinat kartesius

$$y = \frac{x^3}{y^2 - 2x}$$

b. bentuk koordinat kutub

$$f(r,\theta) = r^3\cos^3(\theta) - r^3\sin^3(\theta) + 2r^2\cos(\theta))(\sin(\theta)$$

c. bentuk persamaan parametrik

$$x = r\cos t$$

$$y = r\sin t$$

d. bentuk persamaan implisit

$$f(x,y) = x^3 - y^3 + 2xy = 0$$

Kurvatur koordinat kartesius

```
>function f(x) &= (x^3)/(y^2-2*x)
```

```
Maxima said:
expt: undefined: 0 to a negative exponent.
 -- an error. To debug this try: debugmode(true);

Error in:
function f(x) &= (x^3)/(y^2-2*x) ...
                                 ^
```

```
>function k(x) &= (diff(f(x),x,2))/(1+diff(f(x),x)^2)^(3/2); $'k(x)=k(x)
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
... (x) &= (diff(f(x),x,2))/(1+diff(f(x),x)^2)^(3/2); $'k(x)=k(x) ...
                                                     ^
```

Kurvatur persamaan implisit

```
>function f(x,y) &=x^3-y^3+2*x*y; $'f(x,y)=f(x,y)
```

```
>fx &= diff(f(x,y),x), fxx &=diff(f(x,y),x,2), fy &=diff(f(x,y),y), fxy &=d
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
fx &= diff(f(x,y),x), fxx &=diff(f(x,y),x,2), fy &=diff(f(x,y) ...
                    ^
```

```
>function k(x) &= (fy^2*fxx-2*fx*fy*fxy+fx^2*fyy)/(fx^2+fy^2)^(3/2); $'k(x)
```

$$k\left(\left[0, 1.666665833335744 \times 10^{-7}\,r, 1.333306666920222 \times 10^{-6}\,r, 4.499797504338432 \times 10^{-6}\,r, 1.066581336583\right.\right.$$

Barisan dan Deret

(Catatan: bagian ini belum lengkap. Anda dapat membaca contoh-contoh pengguanaan EMT dan Maxima untuk menghitung limit barisan, rumus jumlah parsial suatu deret, jumlah tak hingga suatu deret konvergen, dan sebagainya. Anda dapat mengeksplor contoh-contoh di EMT atau perbagai panduan penggunaan Maxima di software Maxima atau dari Internet.)

Barisan dapat didefinisikan dengan beberapa cara di dalam EMT, di antaranya:

- dengan cara yang sama seperti mendefinisikan vektor dengan elemen-elemen beraturan (menggunakan titik dua ":");
- menggunakan perintah "sequence" dan rumus barisan (suku ke -n);
- menggunakan perintah "iterate" atau "niterate";
- menggunakan fungsi Maxima "create_list" atau "makelist" untuk menghasilkan barisan simbolik;
- menggunakan fungsi biasa yang inputnya vektor atau barisan;
- menggunakan fungsi rekursif.

EMT menyediakan beberapa perintah (fungsi) terkait barisan, yakni:

- sum: menghitung jumlah semua elemen suatu barisan
- cumsum: jumlah kumulatif suatu barisan
- differences: selisih antar elemen-elemen berturutan

EMT juga dapat digunakan untuk menghitung jumlah deret berhingga maupun deret tak hingga, dengan menggunakan perintah (fungsi) "sum". Perhitungan dapat dilakukan secara numerik maupun simbolik dan eksak.

Berikut adalah beberapa contoh perhitungan barisan dan deret menggunakan EMT.

```
>1:10 // barisan sederhana
```

```
[1,  2,  3,  4,  5,  6,  7,  8,  9,  10]
```

```
>1:2:30
```

```
[1,  3,  5,  7,  9,  11,  13,  15,  17,  19,  21,  23,  25,  27,  29]
```

Iterasi dan Barisan

EMT menyediakan fungsi iterate("g(x)", x0, n) untuk melakukan iterasi

$$x_{k+1} = g(x_k),\ x_0 = x_0, k = 1, 2, 3, ..., n.$$

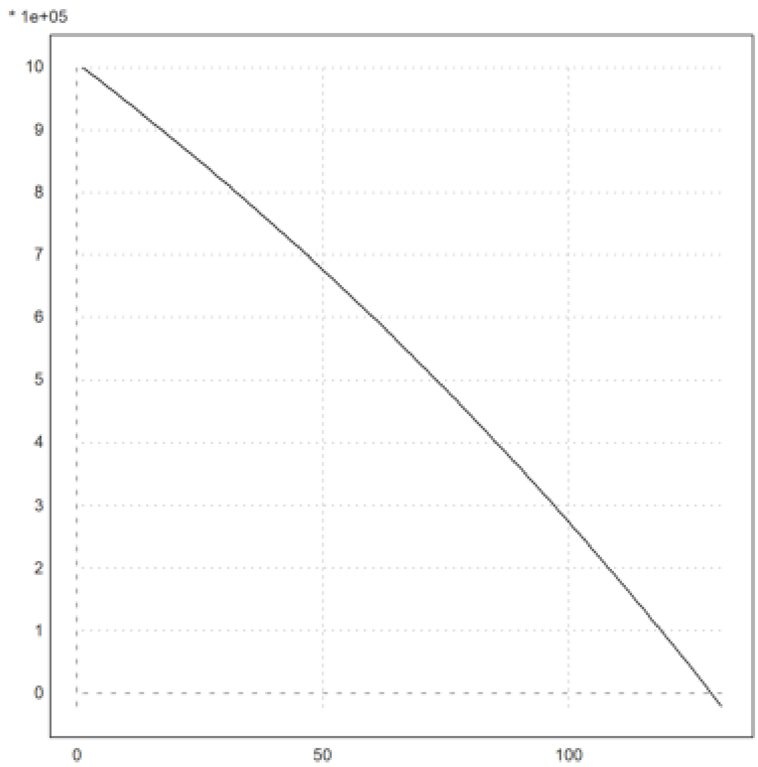
Berikut ini disajikan contoh-contoh penggunaan iterasi dan rekursi dengan EMT. Contoh pertama menunjukkan pertumbuhan dari nilai awal 1000 dengan laju pertambahan 5%, selama 10 periode.

```
>q=1.05; iterate("x*q",1000,n=10)'
```

```
       1000
       1050
     1102.5
    1157.63
    1215.51
    1276.28
     1340.1
     1407.1
    1477.46
    1551.33
    1628.89
```

Contoh berikutnya memperlihatkan bahaya menabung di bank pada masa sekarang! Dengan bunga tabungan sebesar 6% per tahun atau 0.5% per bulan dipotong pajak 20%, dan biaya administrasi 10000 per bulan, tabungan sebesar 1 juta tanpa diambil selama sekitar 10 tahunan akan habis diambil oleh bank!

```
>r=0.005; plot2d(iterate("(1+0.8*r)*x-10000",1000000,n=130)):
```



Silakan Anda coba-coba, dengan tabungan minimal berapa agar tidak akan habis diambil oleh bank dengan ketentuan bunga dan biaya administrasi seperti di atas.

Berikut adalah perhitungan minimal tabungan agar aman di bank dengan bunga sebesar r dan biaya administrasi a, pajak bunga 20%.

```
>$solve(0.8*r*A-a,A), $% with [r=0.005, a=10]
```

$$\left[A = \frac{5\,a}{4\,r}\right]$$

$$[A = 2500.0]$$

Berikut didefinisikan fungsi untuk menghitung saldo tabungan, kemudian dilakukan iterasi.

```
>function saldo(x,r,a) := round((1+0.8*r)*x-a,2);
>iterate({{"saldo",0.005,10}},1000,n=6)
```

```
[1000,  994,  987.98,  981.93,  975.86,  969.76,  963.64]
```

```
>iterate({{"saldo",0.005,10}},2000,n=6)
```

```
[2000,  1998,  1995.99,  1993.97,  1991.95,  1989.92,  1987.88]
```

```
>iterate({{"saldo",0.005,10}},2500,n=6)
```

```
[2500,  2500,  2500,  2500,  2500,  2500,  2500]
```

Tabungan senilai 2,5 juta akan aman dan tidak akan berubah nilai (jika tidak ada penarikan), sedangkan jika tabungan awal kurang dari 2,5 juta, lama kelamaan akan berkurang meskipun tidak pernah dilakukan penarikan uang tabungan.

```
>iterate({{"saldo",0.005,10}},3000,n=6)
```

```
[3000,  3002,  3004.01,  3006.03,  3008.05,  3010.08,  3012.12]
```

Tabungan yang lebih dari 2,5 juta baru akan bertambah jika tidak ada penarikan.

Untuk barisan yang lebih kompleks dapat digunakan fungsi "sequence()". Fungsi ini menghitung nilai-nilai x[n] dari semua nilai sebelumnya, x[1],...,x[n-1] yang diketahui.
Berikut adalah contoh barisan Fibonacci.

$$x_n = x_{n-1} + x_{n-2}, \quad x_1 = 1, \quad x_2 = 1$$

```
>sequence("x[n-1]+x[n-2]",[1,1],15)
```

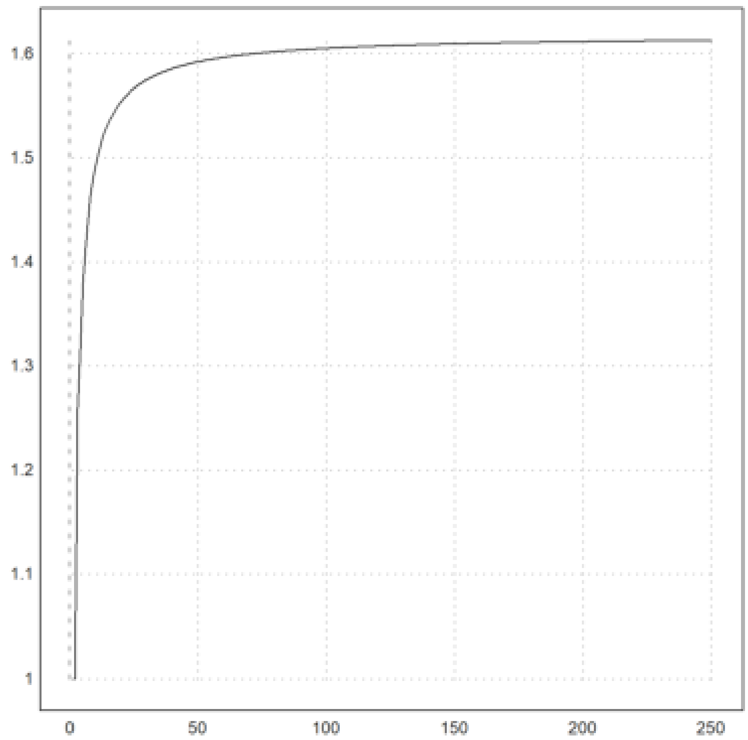
```
[1,  1,  2,  3,  5,  8,  13,  21,  34,  55,  89,  144,  233,  377,  610]
```

Barisan Fibonacci memiliki banyak sifat menarik, salah satunya adalah akar pangkat ke-n suku ke-n akan konvergen ke pecahan emas:

```
>$'(1+sqrt(5))/2=float((1+sqrt(5))/2)
```

$$\frac{\sqrt{5}+1}{2} = 1.618033988749895$$

```
>plot2d(sequence("x[n-1]+x[n-2]",[1,1],250)^(1/(1:250))):
```



Barisan yang sama juga dapat dihasilkan dengan menggunakan loop.

```
>x=ones(500); for k=3 to 500; x[k]=x[k-1]+x[k-2]; end;
```

Rekursi dapat dilakukan dengan menggunakan rumus yang tergantung pada semua elemen sebelumnya. Pada contoh berikut, elemen ke-n merupakan jumlah (n-1) elemen sebelumnya, dimulai dengan 1 (elemen ke-1). Jelas, nilai elemen ke-n adalah 2^(n-2), untuk n=2, 4, 5, ...

```
>sequence("sum(x)",1,10)
```

```
[1,  1,  2,  4,  8,  16,  32,  64,  128,  256]
```

Selain menggunakan ekspresi dalam x dan n, kita juga dapat menggunakan fungsi.

Pada contoh berikut, digunakan iterasi

$$x_n = A \cdot x_{n-1},$$

dengan A suatu matriks 2x2, dan setiap x[n] merupakan matriks/vektor 2x1.

```
>A=[1,1;1,2]; function suku(x,n) := A.x[,n-1]
>sequence("suku",[1;1],6)
```

```
Real 2 x 6 matrix

            1            2            5           13      ...
            1            3            8           21      ...
```

Hasil yang sama juga dapat diperoleh dengan menggunakan fungsi perpangkatan matriks "matrixpower()". Cara ini lebih cepat, karena hanya menggunakan perkalian matriks sebanyak log_2(n).

$$x_n = A.x_{n-1} = A^2.x_{n-2} = A^3.x_{n-3} = ... = A^{n-1}.x_1.$$

```
>sequence("matrixpower(A,n).[1;1]",1,6)
```

```
Real 2 x 6 matrix

            1            5           13           34      ...
            1            8           21           55      ...
```

Spiral Theodorus

image: Spiral_of_Theodorus.png
Spiral Theodorus (spiral segitiga siku-siku) dapat digambar secara rekursif. Rumus rekursifnya adalah:

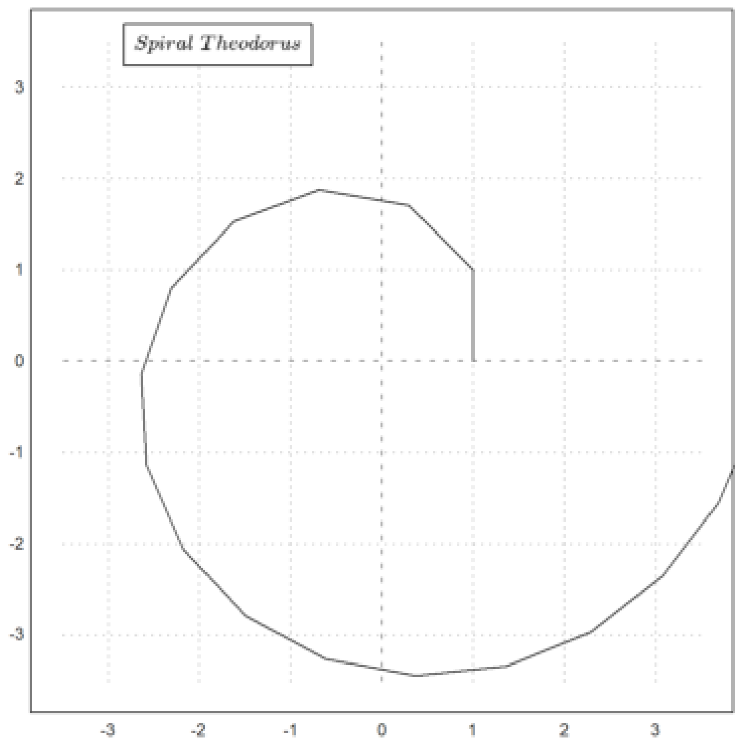
$$x_n = \left(1 + \frac{i}{\sqrt{n-1}}\right) x_{n-1}, \quad x_1 = 1,$$

yang menghasilkan barisan bilangan kompleks.

```
>function g(n) := 1+I/sqrt(n)
```

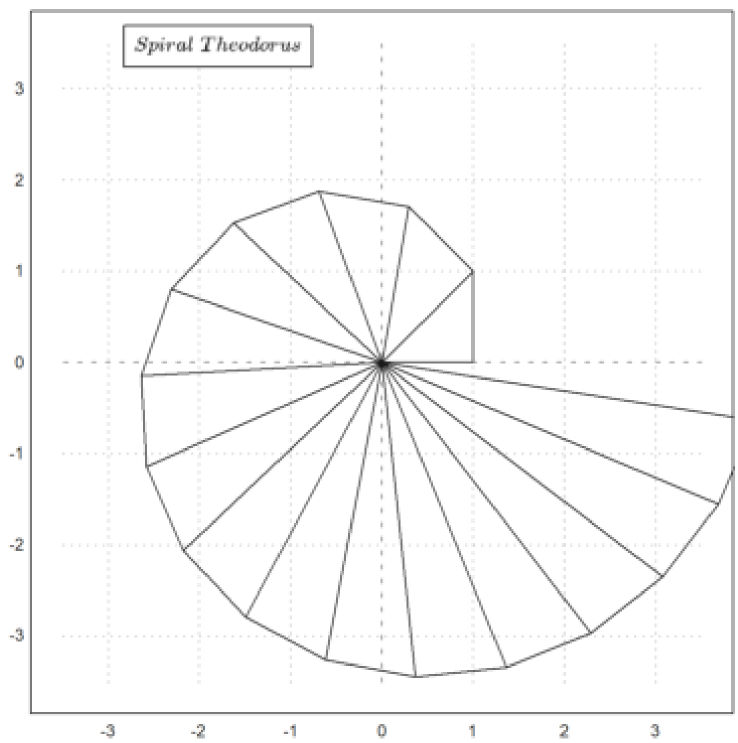
Rekursinya dapat dijalankan sebanyak 17 untuk menghasilkan barisan 17 bilangan kompleks, kemudian digambar bilangan-bilangan kompleksnya.

```
>x=sequence("g(n-1)*x[n-1]",1,17); plot2d(x,r=3.5); textbox(latex("Spiral\
```



Selanjutnya dihubungan titik 0 dengan titik-titik kompleks tersebut menggunakan loop.

```
>for i=1:cols(x); plot2d([0,x[i]],>add); end:
```




```
>
```

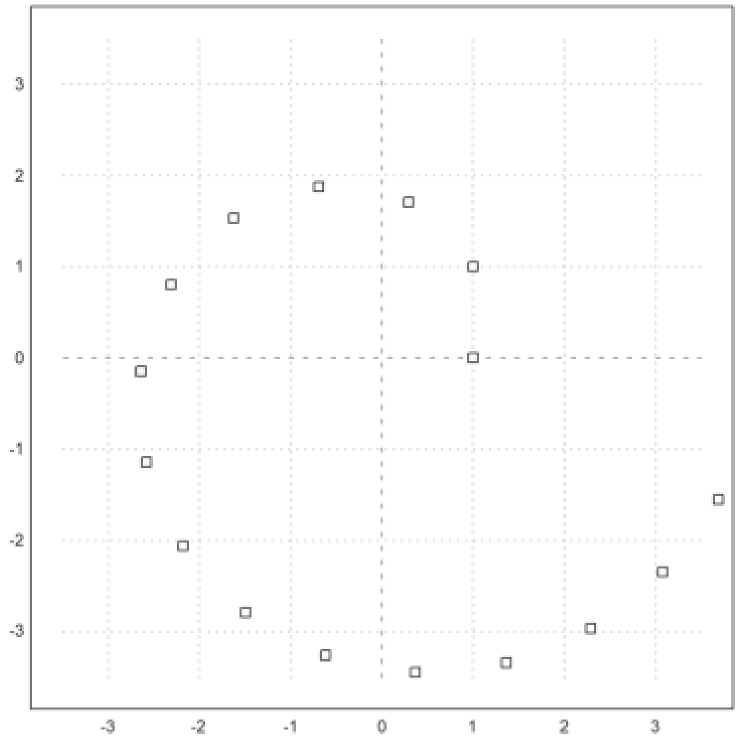
Spiral tersebut juga dapat didefinisikan menggunakan fungsi rekursif, yang tidak memmerlukan indeks dan bilangan kompleks. Dalam hal ini diigunakan vektor kolom pada bidang.

```
>function gstep (v) ...
```

```
  w=[-v[2];v[1]];
  return v+w/norm(w);
  endfunction
```

Jika dilakukan iterasi 16 kali dimulai dari [1;0] akan didapatkan matriks yang memuat vektor-vektor dari setiap iterasi.

```
>x=iterate("gstep",[1;0],16); plot2d(x[1],x[2],r=3.5,>points):
```



Kekonvergenan

Terkadang kita ingin melakukan iterasi sampai konvergen. Apabila iterasinya tidak konvergen setelah ditunggu lama, Anda dapat menghentikannya dengan menekan tombol [ESC].

```
>iterate("cos(x)",1) // iterasi x(n+1)=cos(x(n)), dengan x(0)=1.
```

```
0.739085133216
```

Iterasi tersebut konvergen ke penyelesaian persamaan

$$x = \cos(x).$$

Iterasi ini juga dapat dilakukan pada interval, hasilnya adalah barisan interval yang memuat akar tersebut.

```
>hasil := iterate("cos(x)",~1,2~) //iterasi x(n+1)=cos(x(n)), dengan interv
```

```
~0.739085133211,0.7390851332133~
```

Jika interval hasil tersebut sedikit diperlebar, akan terlihat bahwa interval tersebut memuat akar persamaan x=cos(x).

```
>h=expand(hasil,100), cos(h) << h
```

```
~0.73908513309,0.73908513333~
1
```

Iterasi juga dapat digunakan pada fungsi yang didefinisikan.

```
>function f(x) := (x+2/x)/2
```

Iterasi x(n+1)=f(x(n)) akan konvergen ke akar kuadrat 2.

```
>iterate("f",2), sqrt(2)
```

```
1.41421356237
1.41421356237
```

Jika pada perintah iterate diberikan tambahan parameter n, maka hasil iterasinya akan ditampilkan mulai dari iterasi pertama sampai ke-n.

```
>iterate("f",2,5)
```

```
[2,   1.5,   1.41667,   1.41422,   1.41421,   1.41421]
```

Untuk iterasi ini tidak dapat dilakukan terhadap interval.

```
>niterate("f",~1,2~,5)
```

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

Perhatikan, hasil iterasinya sama dengan interval awal. Alasannya adalah perhitungan dengan interval bersifat terlalu longgar. Untuk meingkatkan perhitungan pada ekspresi dapat digunakan pembagian intervalnya, menggunakan fungsi ieval().

```
>function s(x) := ieval("(x+2/x)/2",x,10)
```

Selanjutnya dapat dilakukan iterasi hingga diperoleh hasil optimal, dan intervalnya tidak semakin mengecil. Hasilnya berupa interval yang memuat akar persamaan:

$$x = \frac{1}{2}\left(x + \frac{2}{x}\right).$$

Satu-satunya solusi adalah

$$x = \sqrt{2}.$$

```
>iterate("s",~1,2~)
```

```
~1.41421356236,1.41421356239~
```

Fungsi "iterate()" juga dapat bekerja pada vektor. Berikut adalah contoh fungsi vektor, yang menghasilkan rata-rata aritmetika dan rata-rata geometri.

$$(a_{n+1}, b_{n+1}) = \left(\frac{a_n + b_n}{2}, \sqrt{a_n b_n}\right)$$

Iterasi ke-n disimpan pada vektor kolom x[n].

```
>function g(x) := [(x[1]+x[2])/2;sqrt(x[1]*x[2])]
```

Iterasi dengan menggunakan fungsi tersebut akan konvergen ke rata-rata aritmetika dan geometri dari nilai-nilai awal.

```
>iterate("g",[1;5])
```

```
        2.60401
        2.60401
```

Hasil tersebut konvergen agak cepat, seperti kita cek sebagai berikut.

```
>iterate("g",[1;5],4)
```

```
              1            3      2.61803      2.60403      2.60401
              5      2.23607      2.59002      2.60399      2.60401
```

Iterasi pada interval dapat dilakukan dan stabil, namun tidak menunjukkan bahwa limitnya pada batas-batas yang dihitung.

```
>iterate("g",[~1~;~5~],4)
```

```
Interval 2 x 5 matrix

~0.99999999999999778,1.00000000000000022~     ...
~4.99999999999999911,5.00000000000000089~     ...
```

Iterasi berikut konvergen sangat lambat.

$$x_{n+1} = \sqrt{x_n}.$$

```
>iterate("sqrt(x)",2,10)
```

```
[2,  1.41421,  1.18921,  1.09051,  1.04427,  1.0219,  1.01089,
1.00543,  1.00271,  1.00135,  1.00068]
```

Kekonvergenan iterasi tersebut dapat dipercepatdengan percepatan Steffenson:

```
>steffenson("sqrt(x)",2,10)
```

```
[1.04888,  1.00028,  1,  1]
```

Iterasi menggunakan Loop yang ditulis Langsung

Berikut adalah beberapa contoh penggunaan loop untuk melakukan iterasi yang ditulis langsung pada baris perintah.

```
>x=2; repeat x=(x+2/x)/2; until x^2~=2; end; x,
```

```
1.41421356237
```

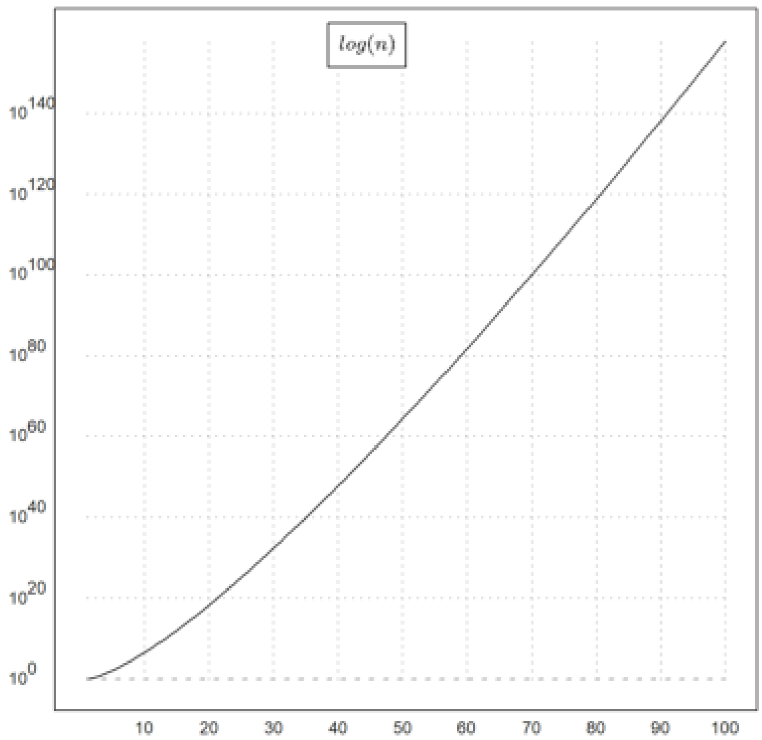
Penggabungan matriks menggunakan tanda "|" dapat digunakan untuk menyimpan semua hasil iterasi.

```
>v=[1]; for i=2 to 8; v=v|(v[i-1]*i); end; v,
```

```
[1,  2,  6,  24,  120,  720,  5040,  40320]
```

hasil iterasi juga dapat disimpan pada vektor yang sudah ada.

```
>v=ones(1,100); for i=2 to cols(v); v[i]=v[i-1]*i; end; ...
>plot2d(v,logplot=1); textbox(latex(&log(n)),x=0.5):
```




```
>A =[0.5,0.2;0.7,0.1]; b=[2;2]; ...
>x=[1;1]; repeat xnew=A.x-b; until all(xnew~=x); x=xnew; end; ...
>x,
```

```
    -7.09677
    -7.74194
```

Iterasi di dalam Fungsi

Fungsi atau program juga dapat menggunakan iterasi dan dapat digunakan untuk melakukan iterasi. Berikut adalah beberapa contoh iterasi di dalam fungsi.

Contoh berikut adalah suatu fungsi untuk menghitung berapa lama suatu iterasi konvergen. Nilai fungsi tersebut adalah hasil akhir iterasi dan banyak iterasi sampai konvergen.

```
>function map hiter(f$,x0) ...
```

```
x=x0;
maxiter=0;
repeat
  xnew=f$(x);
  maxiter=maxiter+1;
  until xnew~=x;
  x=xnew;
end;
return maxiter;
endfunction
```

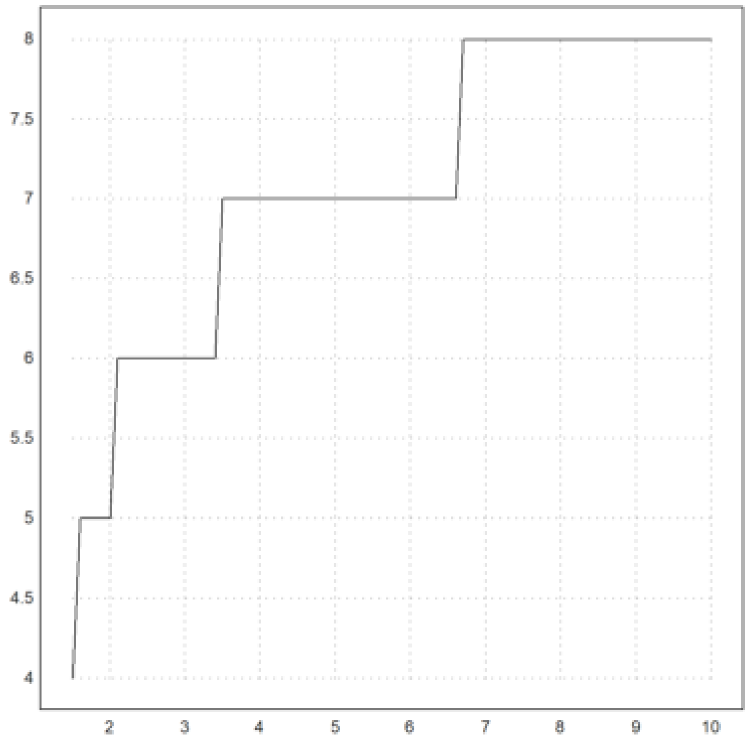
Misalnya, berikut adalah iterasi untuk mendapatkan hampiran akar kuadrat 2, cukup cepat, konvergen pada iterasi ke-5, jika dimulai dari hampiran awal 2.

```
>hiter("(x+2/x)/2",2)
```

```
5
```

Karena fungsinya didefinisikan menggunakan "map". maka nilai awalnya dapat berupa vektor.

```
>x=1.5:0.1:10; hasil=hiter("(x+2/x)/2",x); ...
>  plot2d(x,hasil):
```



Dari gambar di atas terlihat bahwa kekonvergenan iterasinya semakin lambat, untuk nilai awal semakin besar, namun penambahnnya tidak kontinu. Kita dapat menemukan kapan maksimum iterasinya bertambah.

```
>hasil[1:10]
```

```
[4,  5,  5,  5,  5,  5,  6,  6,  6,  6]
```

```
>x[nonzeros(differences(hasil))]
```

```
[1.5,  2,  3.4,  6.6]
```

maksimum iterasi sampai konvergen meningkat pada saat nilai awalnya 1.5, 2, 3.4, dan 6.6.
Contoh berikutnya adalah metode Newton pada polinomial kompleks berderajat 3.

```
>p &= x^3-1; newton &= x-p/diff(p,x); $newton
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
p &= x^3-1; newton &= x-p/diff(p,x); $newton ...
                                    ^
```

Selanjutnya didefinisikan fungsi untuk melakukan iterasi (aslinya 10 kali).

```
>function iterasi(f$,x,n=10) ...
```

```
loop 1 to n; x=f$(x); end;
return x;
endfunction
```

Kita mulai dengan menentukan titik-titik grid pada bidang kompleksnya.

```
>r=1.5; x=linspace(-r,r,501); Z=x+I*x'; W=iterasi(newton,Z);
```

```
Function newton needs at least 3 arguments!
Use: newton (f$: call, df$: call, x: scalar complex {, y: number, eps: no
Error in:
... x=linspace(-r,r,501); Z=x+I*x'; W=iterasi(newton,Z); ...
                                                       ^
```

Berikut adalah akar-akar polinomial di atas.

```
>z=&solve(p)()
```

```
Maxima said:
solve: more equations than unknowns.
Unknowns given :
[r]
Equations given:
errexp1
 -- an error. To debug this try: debugmode(true);

Error in:
z=&solve(p)() ...
           ^
```

Untuk menggambar hasil iterasinya, dihitung jarak dari hasil iterasi ke-10 ke masing-masing akar, kemudian digunakan untuk menghitung warna yang akan digambar, yang menunjukkan limit untuk masing-masing nilai awal.

Fungsi plotrgb() menggunakan jendela gambar terkini untuk menggambar warna RGB sebagai matriks.

```
>C=rgb(max(abs(W-z[1]),1),max(abs(W-z[2]),1),max(abs(W-z[3]),1)); ...
>  plot2d(none,-r,r,-r,r); plotrgb(C):
```

```
Variable W not found!
Error in:
C=rgb(max(abs(W-z[1]),1),max(abs(W-z[2]),1),max(abs(W-z[3]),1) ...
                    ^
```

Iterasi Simbolik

Seperti sudah dibahas sebelumnya, untuk menghasilkan barisan ekspresi simbolik dengan Maxima dapat digunakan fungsi makelist().

```
>&powerdisp:true // untuk menampilkan deret pangkat mulai dari suku berpang
```

```
                                    true
```

```
>deret &= makelist(taylor(exp(x),x,0,k),k,1,3); $deret // barisan deret Tay
```

```
Maxima said:
taylor: 0.1539740213994798*r cannot be a variable.
 -- an error. To debug this try: debugmode(true);

Error in:
deret &= makelist(taylor(exp(x),x,0,k),k,1,3); $deret // baris ...
                                                ^
```

Untuk mengubah barisan deret tersebut menjadi vektor string di EMT digunakan fungsi mxm2str(). Selanjutnya, vektor string/ekspresi hasilnya dapat digambar seperti menggambar vektor eskpresi pada EMT.

```
>plot2d("exp(x)",0,3); // plot fungsi aslinya, e^x
>plot2d(mxm2str("deret"),>add,color=4:6): // plot ketiga deret taylor hampi
```

```
Maxima said:
length: argument cannot be a symbol; found deret
 -- an error. To debug this try: debugmode(true);

mxmeval:
    return evaluate(mxm(s));
Try "trace errors" to inspect local variables after errors.
mxm2str:
    n=mxmeval("length(VVV)");
```

Selain cara di atas dapat juga dengan cara menggunakan indeks pada vektor/list yang dihasilkan.

```
>$deret[3]
```

$$deret_3$$

```
>plot2d(["exp(x)",&deret[1],&deret[2],&deret[3]],0,3,color=1:4):
```

```
deret is not a variable!
Error in expression: deret[1]
%ploteval:
    y0=f$(x[1],args());
```

```
Try "trace errors" to inspect local variables after errors.
plot2d:
    u=u_(%ploteval(xx[#],t;args()));
```

```
>$sum(sin(k*x)/k,k,1,5)
```

$$\ldots 214\, r), \sin(0.\left[0.9206861288690583732335744 \times 10^{-7}\, r) + \frac{\sin(0.184137257730373186667144801276120588667011 (3.999750003628274511560 \ldots}{2} \ldots\right.$$

Berikut adalah cara menggambar kurva

$$y = \sin(x) + \frac{\sin 3x}{3} + \frac{\sin 5x}{5} + \ldots.$$

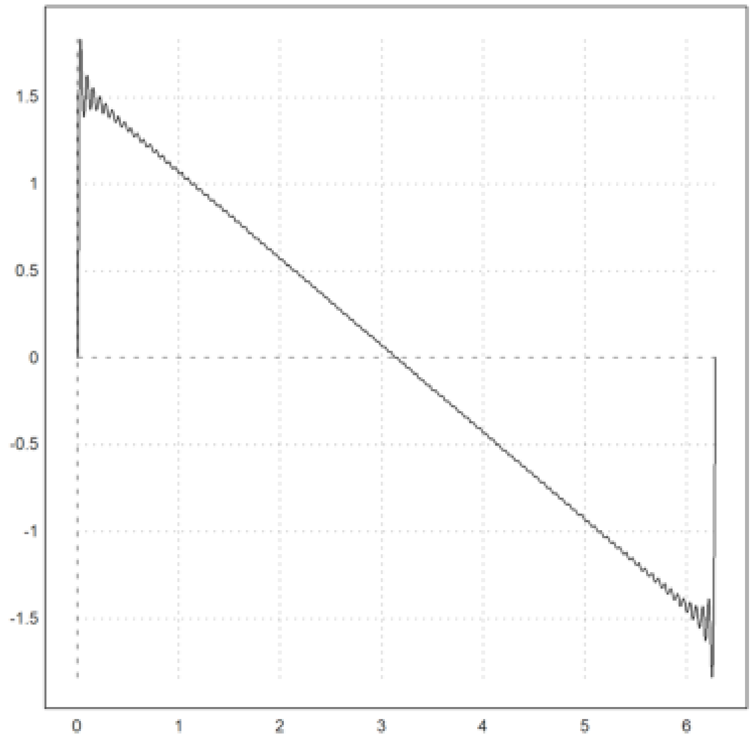
```
>plot2d(&sum(sin((2*k+1)*x)/(2*k+1),k,0,20),0,2pi):
```

```
Maxima output too long!
Error in:
plot2d(&sum(sin((2*k+1)*x)/(2*k+1),k,0,20),0,2pi): ...
                                                  ^
```

Hal serupa juga dapat dilakukan dengan menggunakan matriks, misalkan kita akan menggambar kurva

$$y = \sum_{k=1}^{100} \frac{\sin(kx)}{k}, \quad 0 \leq x \leq 2\pi.$$

```
>x=linspace(0,2pi,1000); k=1:100; y=sum(sin(k*x')/k)'; plot2d(x,y):
```



Tabel Fungsi

Terdapat cara menarik untuk menghasilkan barisan dengan ekspresi Maxima. Perintah mxmtable() berguna untuk menampilkan dan menggambar barisan dan menghasilkan barisan sebagai vektor kolom.

Sebagai contoh berikut adalah barisan turunan ke-n x^x di x=1.

```
>mxmtable("diffat(x^x,x=1,n)","n",1,8,frac=1);
```

```
Maxima said:
diff: second argument must be a variable; found errexp1
#0: diffat(expr=[0,1.666658333357744e-7*r,1.33330666692022e-6*r,4.49979750
 -- an error. To debug this try: debugmode(true);

%mxmevtable:
    return mxm("@expr,@var=@value")();
Try "trace errors" to inspect local variables after errors.
mxmtable:
    y[#,1]=%mxmevtable(expr,var,x[#]);
```

```
>$'sum(k, k, 1, n) = factor(ev(sum(k, k, 1, n),simpsum=true)) // simpsum:me
```

$$\sum_{k=1}^{n}{k}=\frac{n\,\left(1+n\right)}{2}$$

```
>$'sum(1/(3^k+k), k, 0, inf) = factor(ev(sum(1/(3^k+k), k, 0, inf),simpsum=
```

$$\sum_{k=0}^{\infty }{\frac{1}{k+3^{k}}}=\sum_{k=0}^{\infty }{\frac{1}{k+3^{k}}}$$

Di sini masih gagal, hasilnya tidak dihitung.

```
>$'sum(1/x^2, x, 1, inf)= ev(sum(1/x^2, x, 1, inf),simpsum=true) // ev: men
```

$$\sum_{x=1}^{\infty }{\frac{1}{x^2}}=\frac{\pi^2}{6}$$

```
>$'sum((-1)^(k-1)/k, k, 1, inf) = factor(ev(sum((-1)^(x-1)/x, x, 1, inf),si
```

$$\sum_{k=1}^{\infty }{\frac{\left(-1\right)^{-1+k}}{k}}=-\sum_{x=1}^{\infty }{\frac{\left(-1\right)^{x}}{x}}$$

Di sini masih gagal, hasilnya tidak dihitung.

```
>$'sum((-1)^k/(2*k-1), k, 1, inf) = factor(ev(sum((-1)^k/(2*k-1), k, 1, inf
```

$$\sum_{k=1}^{\infty }{\frac{\left(-1\right)^{k}}{-1+2\,k}}=\sum_{k=1}^{\infty }{\frac{\left(-1\right)^{k}}{-1+2\,k}}$$

```
>$ev(sum(1/n!, n, 0, inf),simpsum=true)
```

$$\sum_{n=0}^{\infty }{\frac{1}{n!}}$$

Di sini masih gagal, hasilnya tidak dihitung, harusnya hasilnya e.

```
>&assume(abs(x)<1); $'sum(a*x^k, k, 0, inf)=ev(sum(a*x^k, k, 0, inf),simpsu
```

```
Answering "Is -94914474571+15819*r positive, negative or zero?" with "pos
Maxima said:
sum: sum is divergent.
 -- an error. To debug this try: debugmode(true);

Error in:
... k, 0, inf)=ev(sum(a*x^k, k, 0, inf),simpsum=true), &forget(abs ...
                                                      ^
```

Deret geometri tak hingga, dengan asumsi rasional antara -1 dan 1.

```
>$'sum(x^k/k!,k,0,inf)=ev(sum(x^k/k!,k,0,inf),simpsum=true)
```

$$\left[0, \sum_{k=0}^{\infty} \frac{\left(1.666658333335744 \times 10^{-7}\right)^k r^k}{k!}, \sum_{k=0}^{\infty} \frac{\left(1.333306666692022 \times 10^{-6}\right)^k r^k}{k!}, \sum_{k=0}^{\infty} \frac{\left(4.499797504338432 \times 10^{\right.}{k!}$$

```
>$limit(sum(x^k/k!,k,0,n),n,inf)
```

$$\left[0, \lim_{n\to\infty} \sum_{k=0}^{n} \frac{\left(1.666658333335744 \times 10^{-7}\right)^k r^k}{k!}, \lim_{n\to\infty} \sum_{k=0}^{n} \frac{\left(1.333306666692022 \times 10^{-6}\right)^k r^k}{k!}, \lim_{n\to\infty} \sum_{k=0}^{n} \frac{(4.49979750}{}\right.$$

```
>function d(n) &= sum(1/(k^2-k),k,2,n); $'d(n)=d(n)
```

$$d(n) = \sum_{k=2}^{n} \frac{1}{-k+k^2}$$

```
>$d(10)=ev(d(10),simpsum=true)
```

$$\sum_{k=2}^{10} \frac{1}{-k+k^2} = \frac{9}{10}$$

```
>$d(100)=ev(d(100),simpsum=true)
```

$$\sum_{k=2}^{100} \frac{1}{-k+k^2} = \frac{99}{100}$$

Deret Taylor

Deret Taylor suatu fungsi f yang diferensiabel sampai tak hingga di sekitar x=a adalah:

$$f(x) = \sum_{k=0}^{\infty} \frac{(x-a)^k f^{(k)}(a)}{k!}.$$

```
>$'e^x =taylor(exp(x),x,0,10) // deret Taylor e^x di sekitar x=0, sampai su
```

```
Maxima said:
taylor: 0.1539740213994798*r cannot be a variable.
 -- an error. To debug this try: debugmode(true);

Error in:
$'e^x =taylor(exp(x),x,0,10) // deret Taylor e^x di sekitar x= ...
                              ^
```

```
>$'log(x)=taylor(log(x),x,1,10)// deret log(x) di sekitar x=1
```

```
Maxima said:
log: encountered log(0).
 -- an error. To debug this try: debugmode(true);

Error in:
$'log(x)=taylor(log(x),x,1,10)// deret log(x) di sekitar x=1 ...
                               ^
```

BAB 6

VISUALISASI DAN PERHITUNGAN GEOMETRI DENGAN EMT

Visualisasi dan Perhitungan Geometri dengan EMT

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

```
>load geometry
```

```
Numerical and symbolic geometry.
```

Fungsi-fungsi Geometri

Fungsi-fungsi untuk Menggambar Objek Geometri:

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

koordinat

```
setPlotRange(r): pusat bidang koordinat (0,0) dan batas-batas
```

sumbu-x dan y adalah -r sd r

```
plotPoint (P, "P"): menggambar titik P dan diberi label "P"
plotSegment (A,B, "AB", d): menggambar ruas garis AB, diberi label
```

"AB" sejauh d

```
plotLine (g, "g", d): menggambar garis g diberi label "g" sejauh d
plotCircle (c,"c",v,d): Menggambar lingkaran c dan diberi label "c"
plotLabel (label, P, V, d): menuliskan label pada posisi P
```

Fungsi-fungsi Geometri Analitik (numerik maupun simbolik):

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

ax+by=c.

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

c2

```
planeThrough(A, B, C):  bidang melalui titik A, B, C
```

Fungsi-fungsi Khusus Untuk Geometri Simbolik:

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

y dengan titik A pada

```
sisi positif (kanan/atas) garis
quad(A,B): kuadrat jarak AB
spread(a,b,c): Spread segitiga dengan panjang sisi-sisi a,b,c, yakni
```

sin(alpha)^2 dengan

```
alpha sudut yang menghadap sisi a.
crosslaw(a,b,c,sa): persamaan 3 quads dan 1 spread pada segitiga
```

dengan panjang sisi a, b, c.

```
triplespread(sa,sb,sc): persamaan 3 spread sa,sb,sc yang memebntuk
```

suatu segitiga

```
doublespread(sa): Spread sudut rangkap Spread 2*phi, dengan
```

sa=sin(phi)^2 spread a.

Contoh 1: Luas, Lingkaran Luar, Lingkaran Dalam Segitiga

Untuk menggambar objek-objek geometri, langkah pertama adalah menentukan rentang sumbu-sumbu koordinat. Semua objek geometri akan digambar pada satu bidang koordinat, sampai didefinisikan bidang koordinat yang baru.

```
>setPlotRange(-0.5,2.5,-0.5,2.5); // mendefinisikan bidang koordinat baru
```

Sekarang tetapkan tiga titik dan gambarkan.

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

Lalu tiga segmen.

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

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

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

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

Hitung garis tegak lurus yang melalui A di BC.

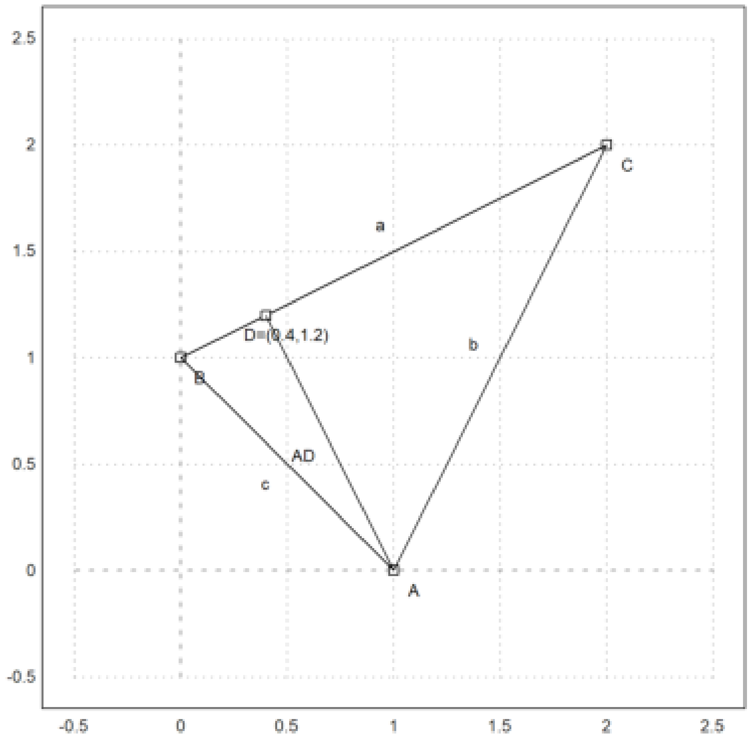
```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan persimpangannya dengan BC.
lalui A di BC.

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

Buatlah grafik itu.

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



Hitung luas ABC:

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

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

```
1.5
```

Bandingkan dengan rumus determinan.

```
A di BC.
```

```
>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
```

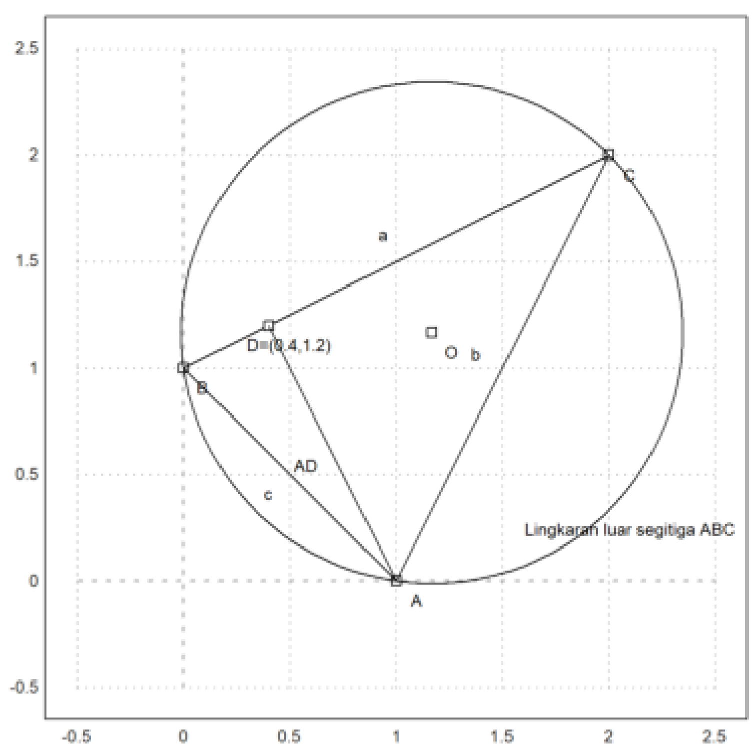
Sudut di C

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

```
36°52'11.63''
```

Sekarang lingkaran luar segitiga.

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



Tampilkan koordinat titik pusat dan jari-jari lingkaran luar.

```
>O, R
```

```
[1.16667,  1.16667]
1.17851130198
```

Sekarang akan digambar lingkaran dalam segitiga ABC. Titik pusat lingkaran dalam adalah titik potong garis-garis bagi sudut.

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

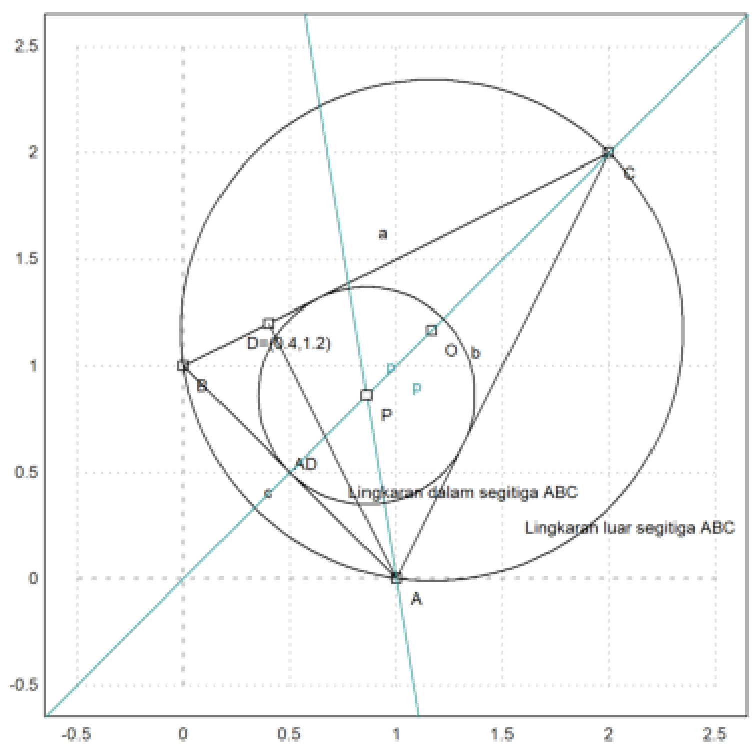
```
[0.86038,  0.86038]
```

Tambahkan semuanya ke plot.

```
>color(5); plotLine(l); plotLine(g); color(1); // gambar kedua garis bagi s
>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"): // gamba
```



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?

Penyelesaian

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

```
>L=circleWithCenter(P,r);
>M1=lineThrough(A,B); //sisi AB pada segitiga ABC
>M2=lineThrough(B,C); //sisi BC pada segitiga ABC
>M3=lineThrough(C,A); //sisi CA pada segitiga ABC
>N1=lineCircleIntersections(M1,L), //titik singgung sisi AB dan lingkaran d
```

```
[0.5,  0.5]
```

```
>N2=lineCircleIntersections(M2,L), //titik singgung sisi BC dan lingkaran d
```

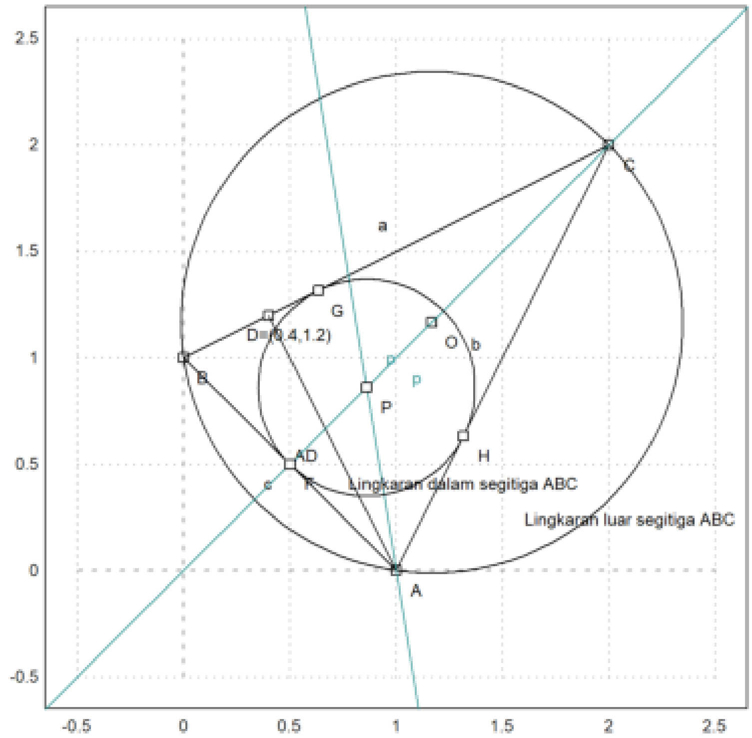
```
[0.632456,  1.31623]
```

```
>N3=lineCircleIntersections(M3,L), //titik singgung sisi CA dan lingkaran d
```

```
[1.31623,  0.632456]
```

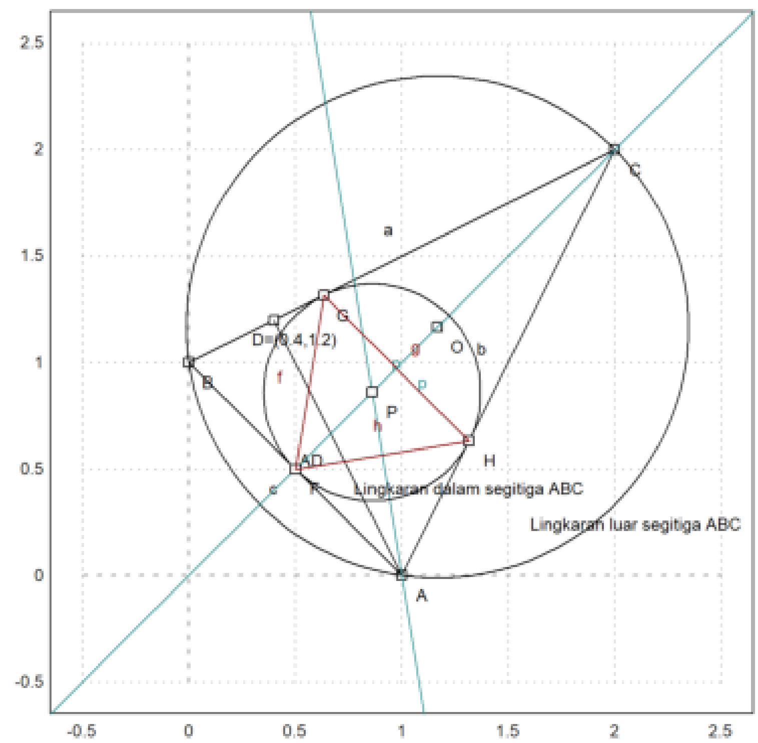
sehingga diperoleh titik singgung sebagai berikut.

```
> plotPoint(N1,"F"); plotPoint(N2,"G"); plotPoint(N3,"H"):
```



2. Akan digambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut. Dan akan dicari tau merupakan segitiga apakah itu?

```
>color(2); plotSegment(N1,N2,"f"); // f=FG
>color(2); plotSegment(N2,N3,"g"); // g=GH
>color(2); plotSegment(N1,N3,"h"): // h=HF
```




```
>norm(N1-N2)//panjang sisi f
```

    0.826905214631

```
>norm(N2-N3)//panjang sisi g
```

    0.966999966873

```
>norm(N1-N3)//panjang sisi h
```

    0.826905214631

Karena segitiga FGH memiliki dua sisi yang sama panjang yaitu sisi f dan h, maka segitiga tersebut adalah segitiga sama kaki.

3. Akan dihitung luas segitiga tersebut.

```
>F=N1
```

    [0.5, 0.5]

```
>G=N2
```

```
[0.632456,  1.31623]
```

```
>H=N3
```

```
[1.31623,  0.632456]
```

```
>areaTriangle(F,G,H) // hitung luas segitiga langsung dengan fungsi
```

```
0.324341649025
```

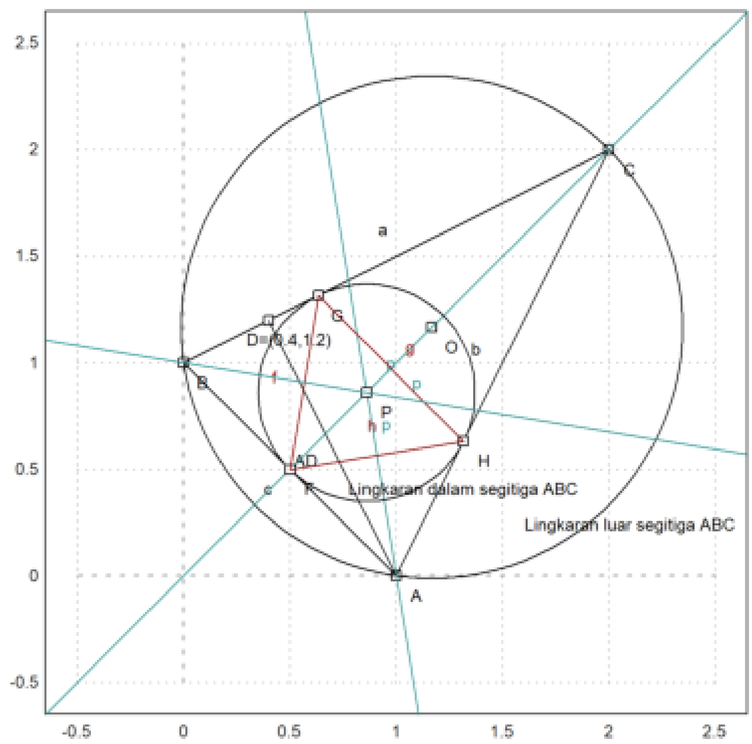
Jadi, luas dari segitiga FGH adalah 0.324341649025.

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

```
>l=angleBisector(A,C,B); // garis bagi <ACB
>k=angleBisector(A,B,C); // garis bagi <ABC
>q=lineIntersection(l,k) // titik potong kedua garis bagi sudut
```

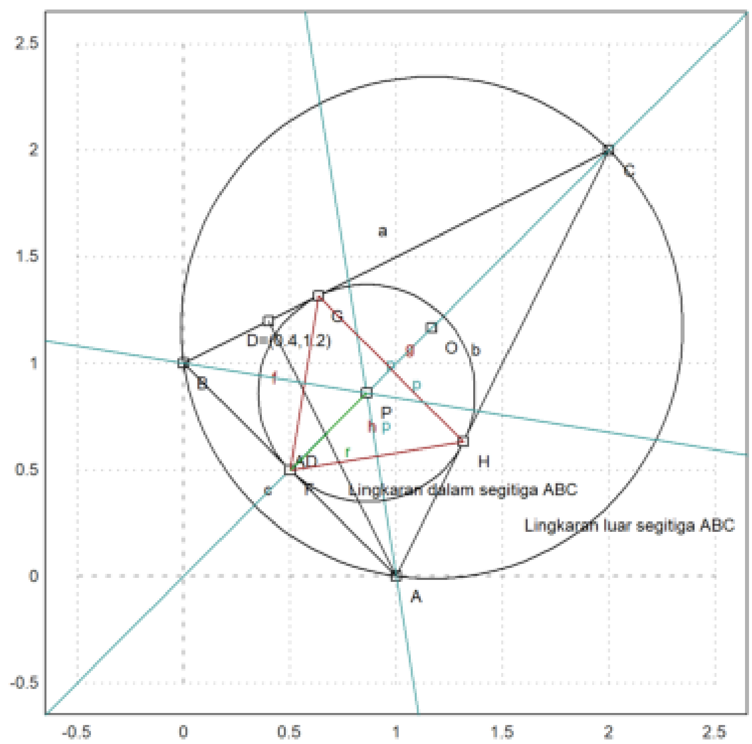
```
[0.86038,  0.86038]
```

```
>color(5); plotLine(k): // gambar garis k yang merupakan garis bagi sudut y
```



Jadi, terlihat bahwa garis bagi sudut yang ke tiga juga melalui titik pusat lingkaran dalam.
5. Akan digambar jari-jari lingkaran dalam.

```
>color(3); plotSegment(P,F,"r"): // r=PF
```



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

```
>ci=circleThrough(F,G,H);
>r=getCircleRadius(ci);
>L1=pi*R^2//luas lingkaran luar
```

```
4.36332312999
```

```
>L2=pi*r^2//luas lingkaran dalam
```

```
0.81601903655
```

```
>areaTriangle(A,B,C)//luas segitiga
```

```
1.5
```

Jadi, hubungan antara luas kedua lingkaran tersebut dengan luas segitia ABC adalah luas lingkaran dalam < luas segitiga < luas lingkaran luar.

Contoh 2: Geometri Smbolik

Kita dapat menghitung geometri eksak dan simbolik menggunakan Maxima.

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

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

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

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

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

Kita bisa mendapatkan persamaan garis dengan mudah.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

```
>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
```

$$y = x$$

```
>P &= lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A)); $P // ti
```

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

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

```
[0.86038,  0.86038]
```

Perpotongan Garis dan Lingkaran

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

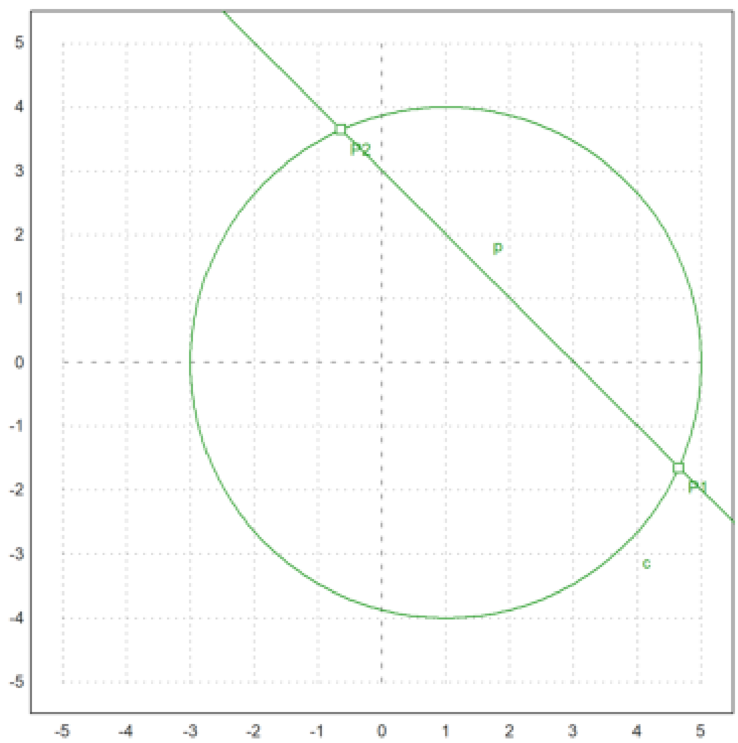
```
>A &:= [1,0]; c=circleWithCenter(A,4);
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);
>setPlotRange(5); plotCircle(c); plotLine(l);
```

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

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

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

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



Hal yang sama di Maxima.

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

[1, 0, 4]

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

[1, 1, 3]

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

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

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

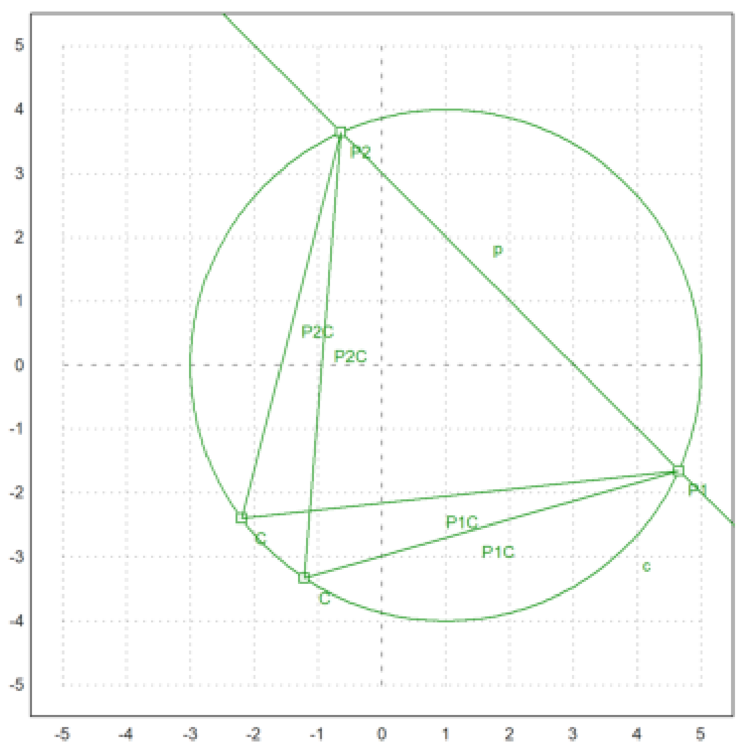
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,
>degprint(computeAngle(P1,C,P2))
```

69°17'42.68''

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

69°17'42.68''

```
>insimg;
```

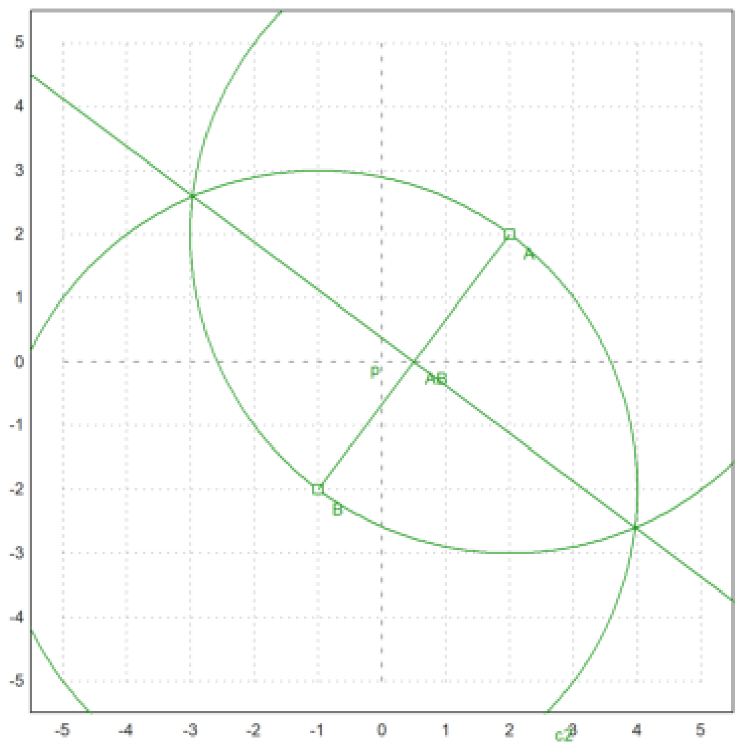


Garis Sumbu

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

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

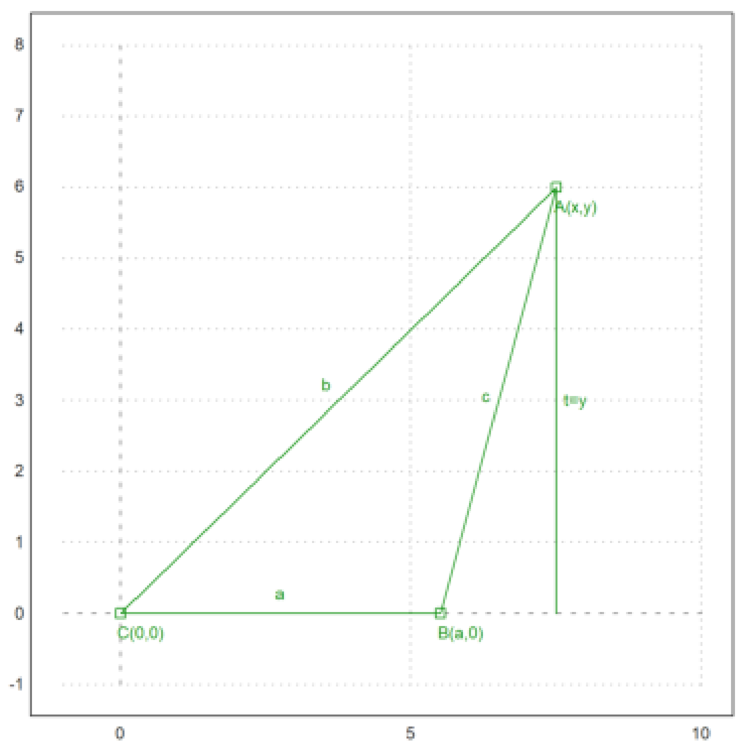
Untuk membuktikan hal ini kita misalkan C(0,0), B(a,0) dan A(x,y), b=AC, c=AB. Luas segitiga ABC adalah

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

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], "
> 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])
```

[]

Extract the solution 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)) ...
                        ^
```

Kami mendapatkan rumus Heron.

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

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

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

$$Luas = |ysol|$$

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

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

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

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

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

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

Kasus umum juga berhasil.

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

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

Error in:
 $solve(diff(H(a,b,c)^2,c)=0,c) ...
                                ^
```

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

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

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

Error in:
s1 &= subst(d-c,b,sol[2]); $s1 ...
                           ^
```

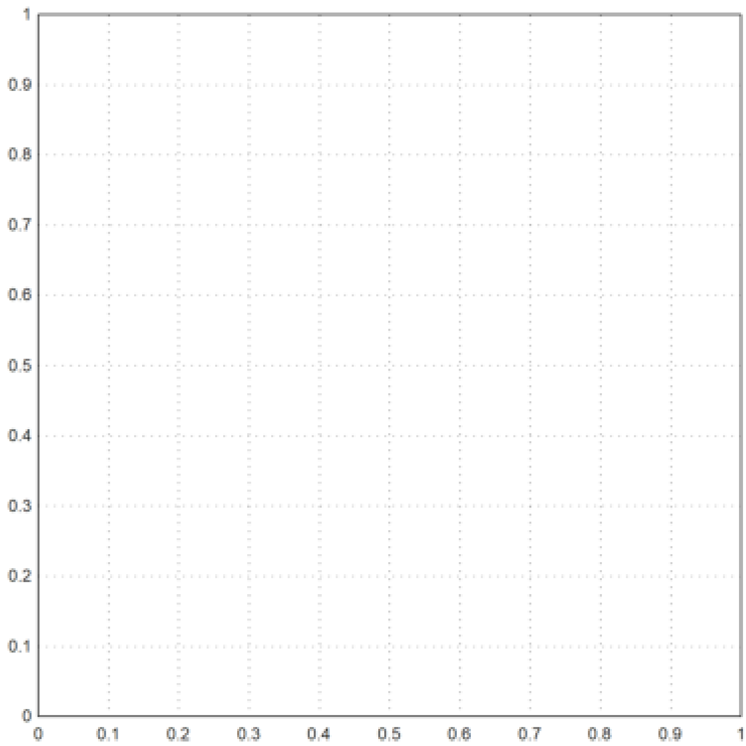
Dan buatlah fungsinya

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

$$0$$

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

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



Kita dapat memeriksa persamaan umum elips ini, yaitu

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

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

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

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

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

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

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

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

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

$$[[a=0,b=\%r_1]]$$

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

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

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

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

Kita bisa membuat plot situasinya

Pertama atur poin di Maxima.

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

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

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

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

$$[a, 0]$$

Kemudian atur rentang plot, dan plot titik-titiknya.

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

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

Plot segmennya.

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

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

Hitung garis tengah tegak lurus di Maxima.

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

Dan pusat lingkarannya.

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

Kita mendapatkan rumus jari-jari lingkaran luar.

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

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

Mari kita tambahkan ini ke dalam plot.

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

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

Dengan menggunakan geometri, kita memperoleh rumus sederhana

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

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

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

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

Contoh 4: Garis Euler dan Parabola

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

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

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

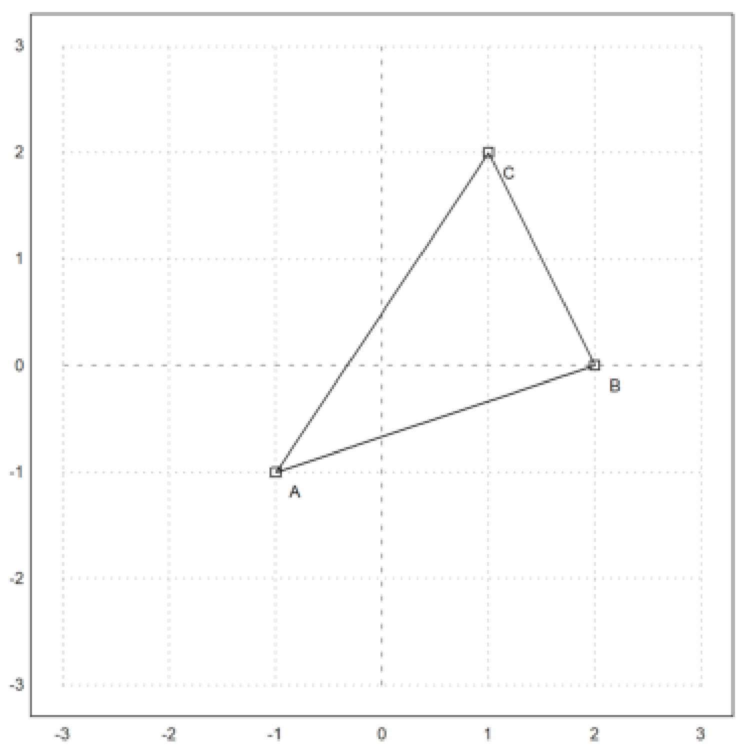
```
>A::=[-1,-1]; B::=[2,0]; C::=[1,2];
```

Untuk memplot objek geometris, kita menyiapkan area plot, dan menambahkan titik ke dalamnua. Semua plot objek geometris ditambahkan ke plot saat ini.

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

Kita juga bisa menjumlahkan sisi-sisi segitiga.

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



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

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

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

Kita dapat menghitung koefisien sisi c.

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

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

Dan dapatkan juga rumus untuk baris ini.

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

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

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

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

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

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

Sekarang kita menghitung lingkaran luar ABC.

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

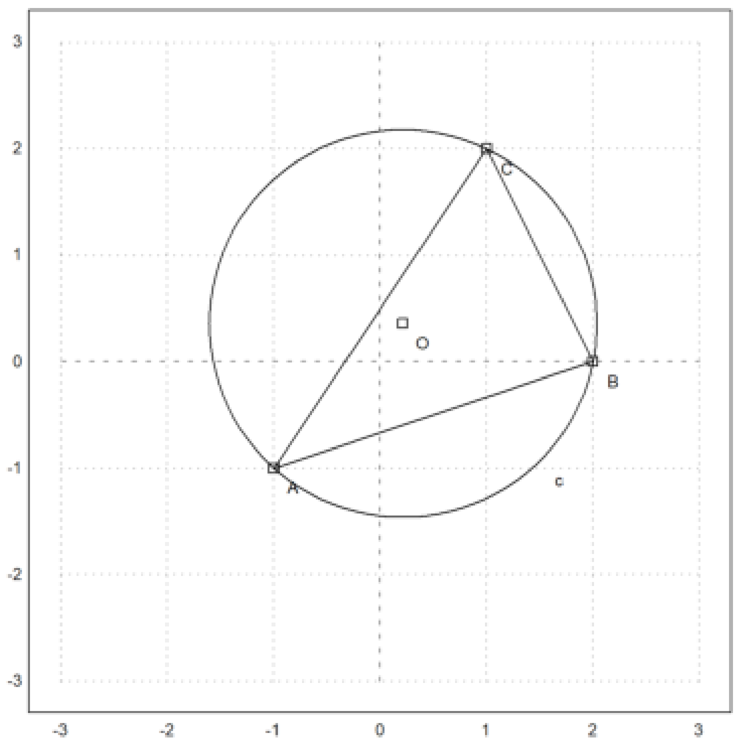
$$\left(y-\frac{5}{14}\right)^2+\left(x-\frac{3}{14}\right)^2=\frac{325}{98}$$

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

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

Plot lingkatan pusatnnya. Cu dan U bersifat simbolis. Kami mengevaluasi ekspresi ini untuk Euler.

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



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

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

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

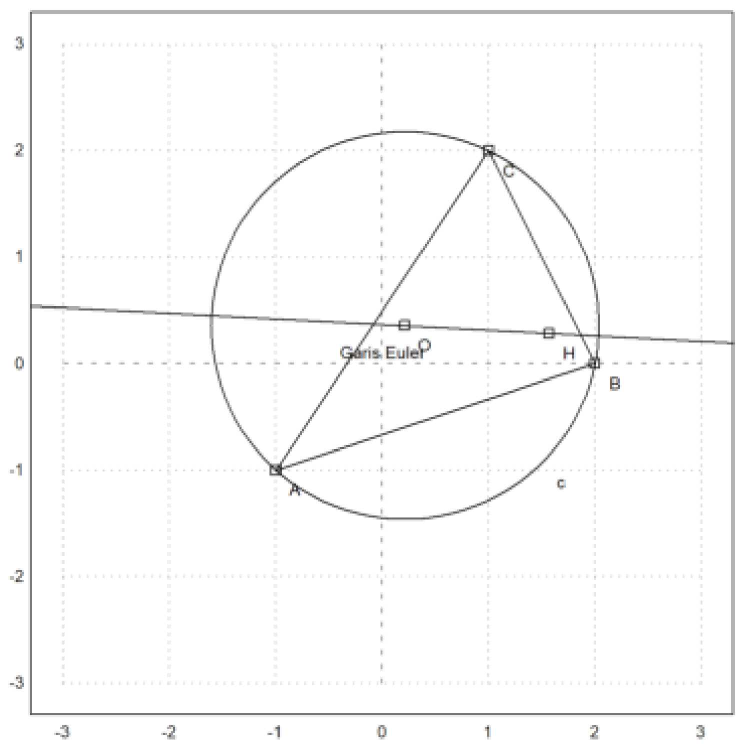
Sekarang kita dapat menghitung garis segitiga Euler.

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

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

Tambahkan ke plot kami.

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

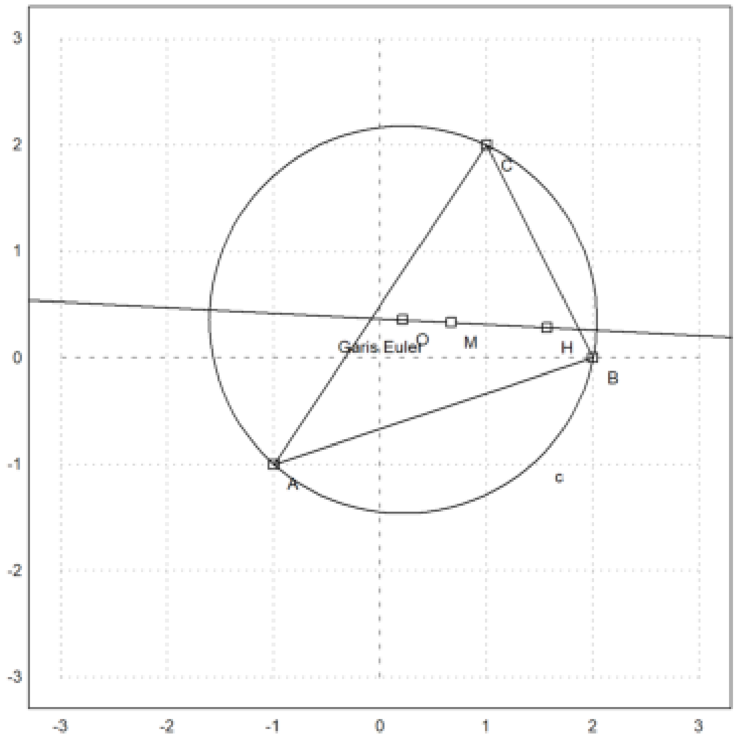


Pusat gravitasi seharusnya berada di garis ini.

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

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

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



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

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

$$2$$

Fungsinya mencakup fungsi untuk sudut juga.

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

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

```
60°15'18.43''
```

Persamaan pusat lingkaran tidak terlalu bagus.

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

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

Mari kita hitung juga ekspresi jari-jari lingkaran yang tertulis.

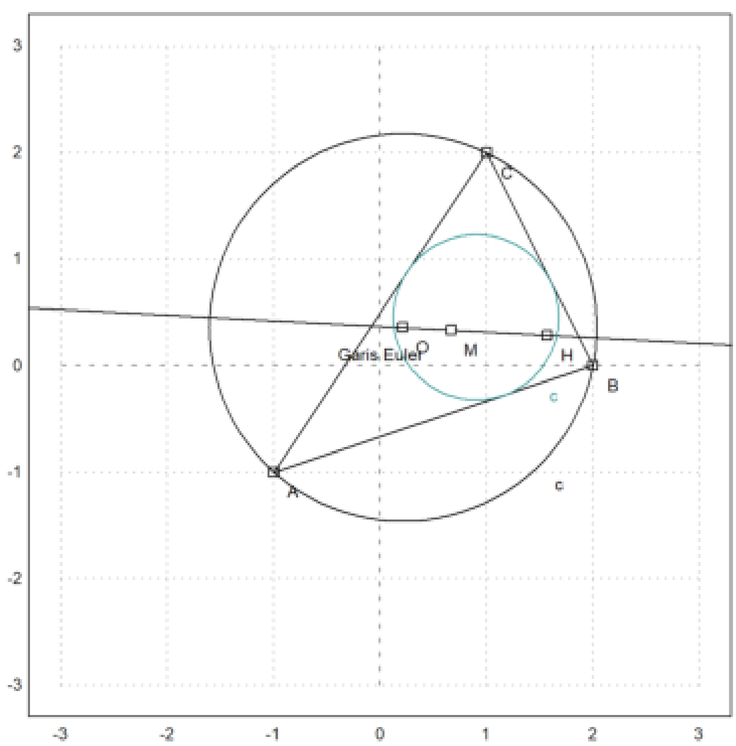
```
>r &= distance(Q,projectToLine(Q,lineThrough(A,B)))|ratsimp; $r
```

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

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

Mari kita tambahkan ini ke dalam plot.

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



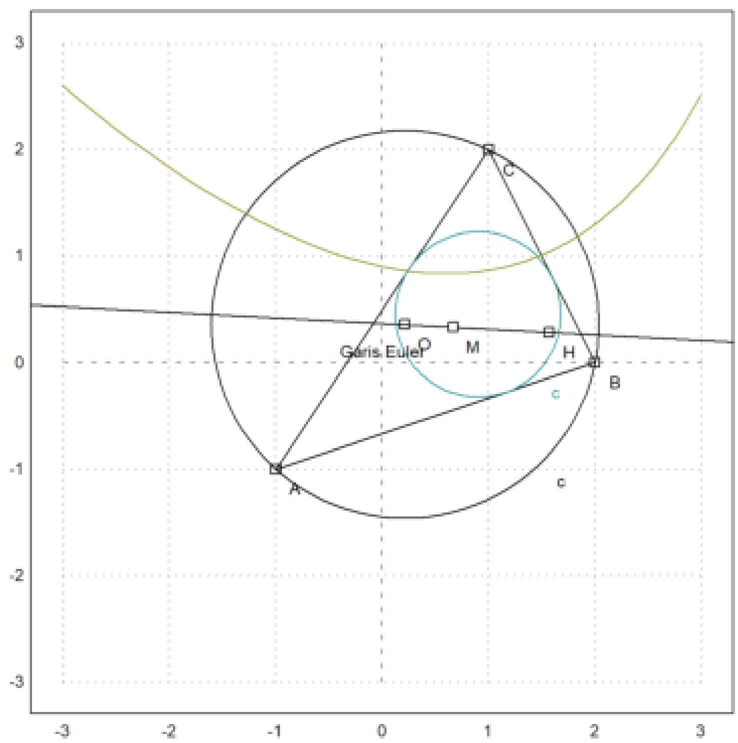
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

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

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

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

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



Ini seharusnya memiliki beberapa fungsi, tetapi pemecah default Maxima hanya dapat menemukan solusinya, jika kita mengkuadratkannya persamaan. Akibatnya, kita mendapatkan solusi palsu.

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

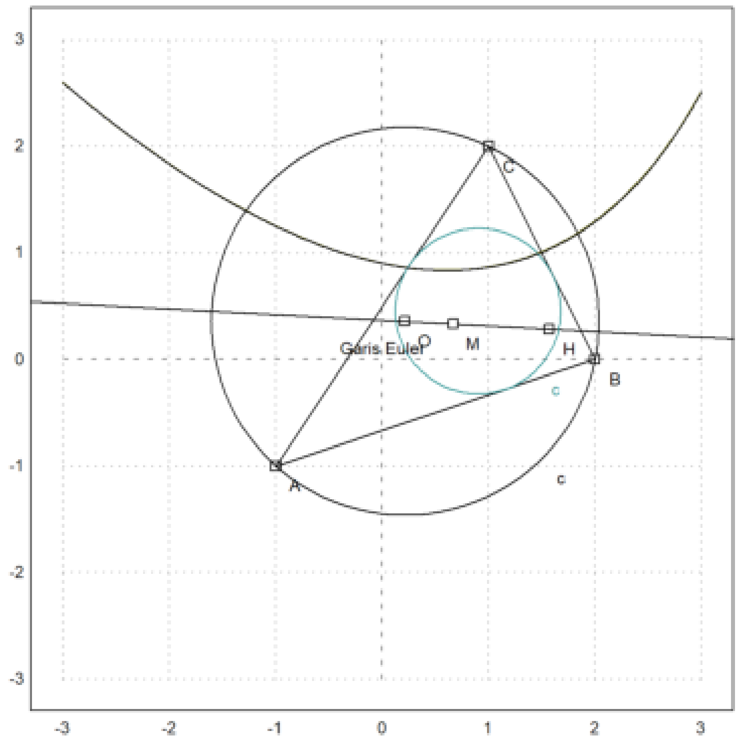
```
[y = - 3 x - sqrt(70) sqrt(9 - 2 x) + 26,
                      y = - 3 x + sqrt(70) sqrt(9 - 2 x) + 26]
```

Solusi pertama adalah

maxima: akar[1]

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

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




```
>function g(x) &= rhs(akar[1]); $'g(x)= g(x)// fungsi yang mendefinisikan k
```

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

```
>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

Hal ini terinspirasi dari perkataan N.J.Wildberger. Dalam bukunya "Divine Proportions", Wildberger mengusulkan untuk menggantikan gagasan klasik tentang jarak dan sudut menurut kuadran dan penyebaran. Dengan menggunakan ini, memang mungkin untuk menghindarinya fungsi trigonometri dalam banyak contoh, dan tetap "rasional".

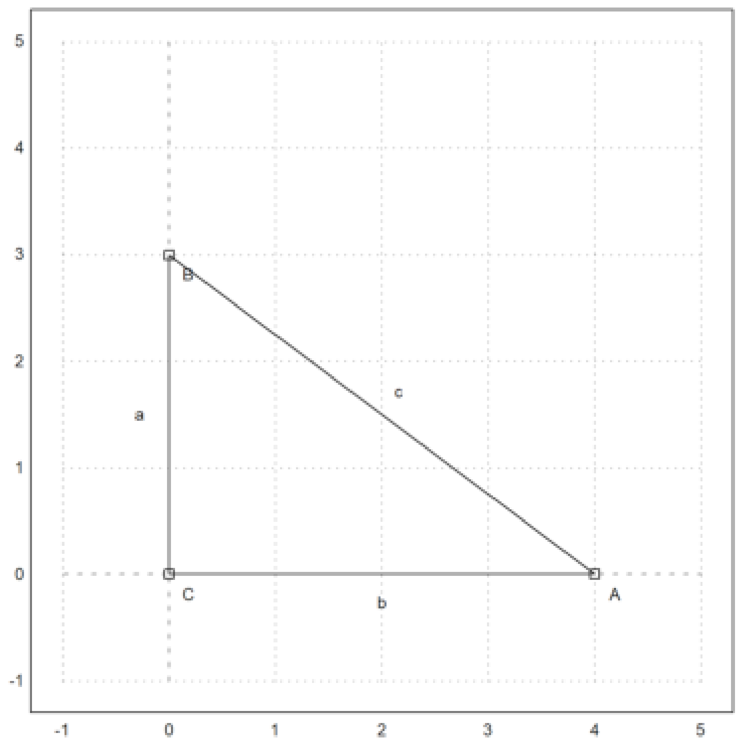
Berikut ini, saya memperkenalkan konsep, dan memecahkan beberapa masalah. Saya menggunakan perhitungan simbolik Maxima di sini, yang menyembunyikan keuntungan utama trigonometri rasional bahwa perhitungannya dapat dilakuan dengan kertas dan pensil saja. Anda diajak untuk memeriksa hasilnya tanpa komputer.

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

```
>load geometry;
```

Untuk pengenalan pertama,kita menggunakan segitiga siku-siku dengan Egyptian proportions 3, 4 and 5. Perintah berikut adalah perintah Euler untuk membuat plot 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,

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

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

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

```
36°52'11.63''
```

Trigonometri rasional mencoba menghindari hal ini.

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

Teorema Pythogoras menjadi a+b=c.

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

```
25 = 25
```

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

Luas garis AB dan AC pada gambar di atas didefinisikan sebagai

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

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

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

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

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

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

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

```
9/25
```

Hukum kosinus trgonometri klasik diterjemahkan menjadi "hukum silang" berikut.

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

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

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

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

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

Dalam kasus kita, kita mendapatkan.

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

$$1024 = 1024$$

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

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

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

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

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

Kami sudah tahu ini. Definisi dari spread adalah kasus khusus dari crosslaw.

Kita juga bisa menyelesaikan ini untuk umum a,b,c. Hasilnya adalah rumus yang menghitung spread dari sudut segitiga yang diberikan quadrances dari tiga sisi.

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

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

Kita bisa membuat fungsi dari hasilnya. Fungsi semacam itu sudah didefinisikan dalam file geometry.e milik Euler.

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

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

Sebagai contoh, kita bisa menggunakannya untuk menghitung sudut dari sebuah segitiga dengan sisi-sisi tersebut.

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

Hasilnya adalah bilangan rasional, yang tidak begitu mudah didapatkan jika kita menggunakan trigonometri klasik.

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

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

Ini adalah sudut dalam derajat.

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

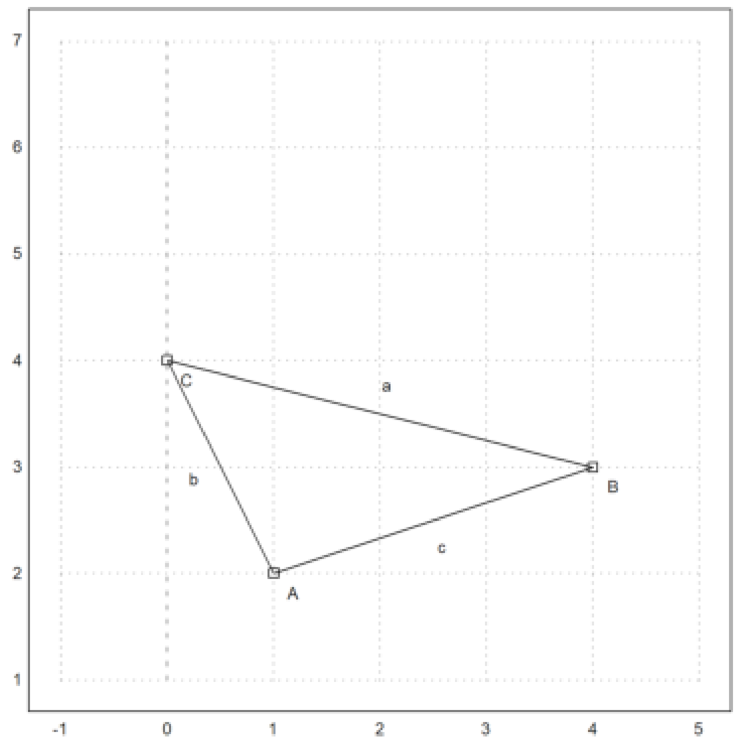
```
67°47'32.44''
```

Contoh Lain

Sekarang, mari mencoba contoh yang lebih kompleks.

Kita atur tiga sudut dari sebuah segitiga sebagai berikut.

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



Dengan menggunakan Pythagoras, mudah untuk menghitung jarak antara dua titik. Saya pertama kali menggunakan fungsi ‘distance‘ dari file Euler untuk geometri. Fungsi ’distance‘ menggunakan geometri klasik.

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

$$\sqrt{10}$$

Euler juga memiliki fungsi untuk menghitung quadrance antara dua titik.

Pada contoh berikut, karena c+b bukan sama dengan a, segitiganya tidak bersifat segitiga siku-siku.

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

17

Pertama, mari hitung sudut tradisional. Fungsi ‘computeAngle‘ menggunakan metode biasa berdasarkan hasil perkalian dot dua vektor. Hasilnya adalah beberapa estimasi titik mengambang.

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

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

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

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

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

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

```
32.4711922908
```

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

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

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

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

Yaitu, 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. Maxima memecahkan persamaan sin(arccos(...)) menjadi hasil pecahan. Sebagian besar mahasiswa mungkin tidak dapat melakukannya.

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

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

Setelah kita memiliki spread di titik B, kita dapat menghitung tinggi ha pada saat a. Ingat bahwa

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

berdasarkan definisi.

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

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

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

image: (20) Rational_Geometry_CaR.png

Berdasarkan definisi, panjang ha adalah akar kuadrat dari quadrance-nya.

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

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

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

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

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

Formula determinan biasa memberikan hasil yang sama.

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

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

Formula Heron

Sekarang, mari selesaikan masalah ini secara umum!

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

Pertama-tama, kita menghitung spread di titik B untuk segitiga dengan sisi-sisi a, b, dan c. Kemudian kita menghitung luas segitiga yang telah dipangkatkan (quadrance area?), mengfaktorkannya dengan Maxima, dan kita akan mendapatkan rumus terkenal Heron.

Mengakui bahwa 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

Kerugiannya adalah bahwa spread tidak lagi hanya menambahkan sudut-sudut seperti sudut biasa.

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 apa pun yang jumlahnya 180°.

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

Karena spread dari

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

sama, aturan triple spread juga benar, jika

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

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

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

Sebagai contoh, kita bisa menghitung spread dari sudut 60°. Hasilnya adalah 3/4. Namun, persamaan-persamaan tersebut memiliki solusi kedua di mana semua spread adalah 0.

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

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

Spread dari 90° jelas adalah 1. Jika dua sudut jumlahnya 90°, spread mereka memenuhi persamaan triple spread dengan a, b, 1. Dengan 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 adalah hasil penjumlahan atau pengurangan dari dua sudut lainnya.

Jadi, kita dapat menemukan spread dari sudut ganda. Perhatikan bahwa ada dua solusi lagi. Kita membuat ini menjadi sebuah fungsi.

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

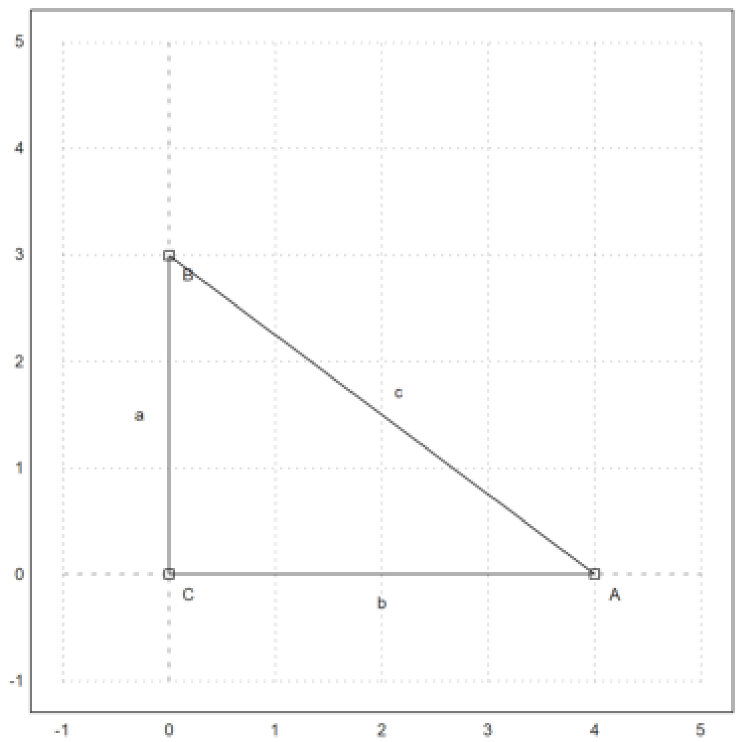
$$\left[x=4\,a-4\,a^2 , x=0\right]$$

```
- 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 menghitung panjang penengah sudut di A. Tetapi kita ingin menyelesaikan ini untuk umum a, b, c.

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

Jadi pertama-tama kita menghitung sebaran sudut yang dibagi di A, menggunakan rumus sebaran tiga sudut.

Masalah dengan rumus ini muncul kembali. Ini memiliki dua solusi. Kita harus memilih yang benar. Solusi lainnya mengacu 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 Mesir.

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

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

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

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

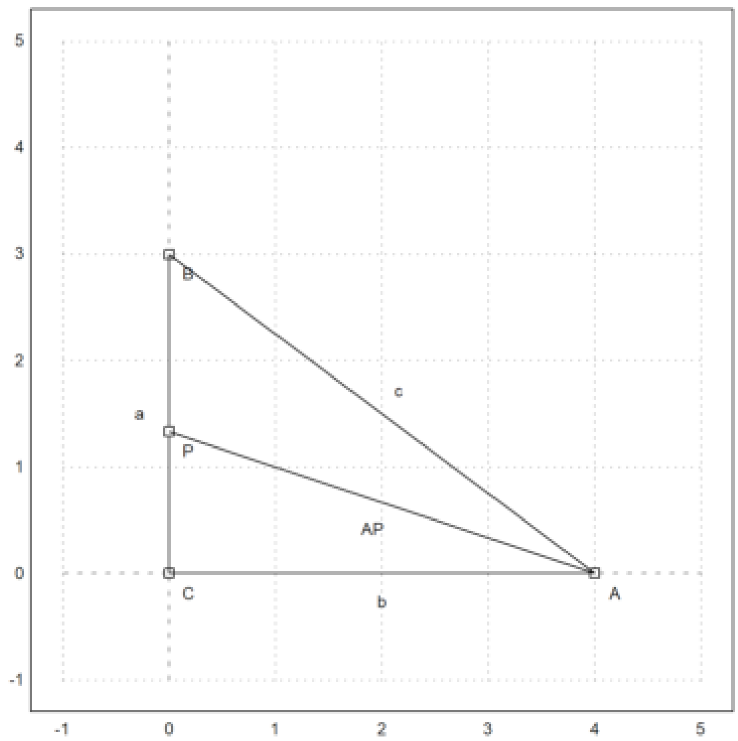
```
18°26'5.82''
```

Titik P adalah perpotongan dari penengah 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 khusus kita.

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

```
0.321750554397
0.321750554397
```

Sekarang kita menghitung panjang penengah 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 apa pun. Kuadratkan, ini berarti menjadi apa yang disebut "spread law"

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

di mana a, b, c mengacu pada kuadrans.

Karena sebaran CPA adalah 1-sa2, kita mendapatkan dari itu bisa 1=b/(1=sa2) dan dapat menghitung bisa (kuadrans dari penengah 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 Egyptian values.

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

```
4.21637021356
4.21637021356
```

Kita juga dapat menghitung P menggunakan rumus sebaran.

```
>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 dengan rumus trigonometri.

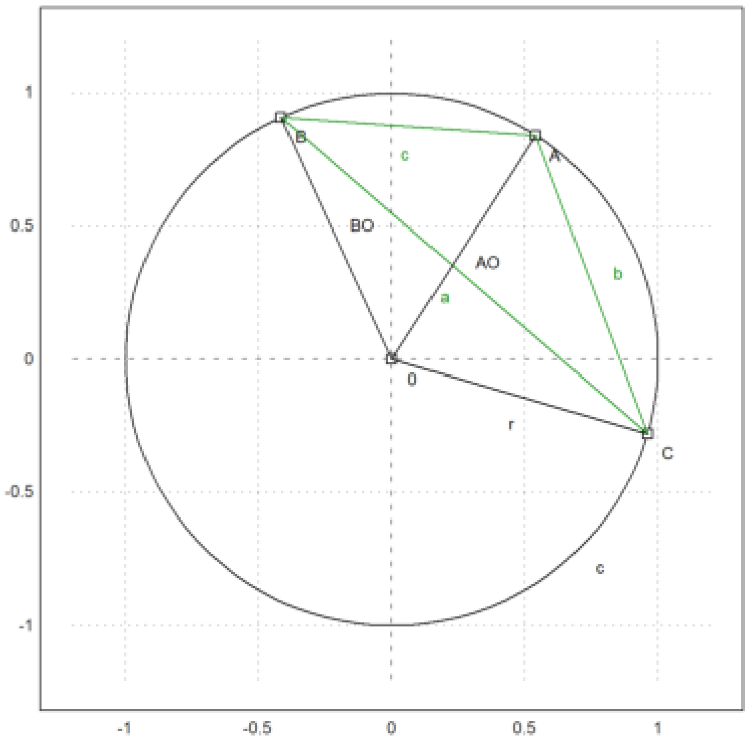
```
>sqrt(mxmeval("at(py,[a=3^2,b=4^2])"))
```

```
1.33333333333
```

Sudut Cincin

Lihat 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 sebaran tiga kali lipat untuk sudut-sudut di pusat O untuk r. Dengan demikian, kita mendapatkan rumus untuk kuadratik radius perisirkel dalam bentuk kuadrans dari sisi-sisi.

Kali ini, Maxima menghasilkan beberapa akar 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)),
```

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

Kita dapat membuatnya menjadi 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 adalah 1.

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

```
1
```

Faktanya adalah bahwa sebaran CBA hanya bergantung pada b dan c. Ini adalah teorema sudut cincin.

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

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

Sebenarnya, sebarannya adalah b/(4r), dan kita melihat bahwa sudut cincin dari tali b adalah separuh dari sudut pusat.

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

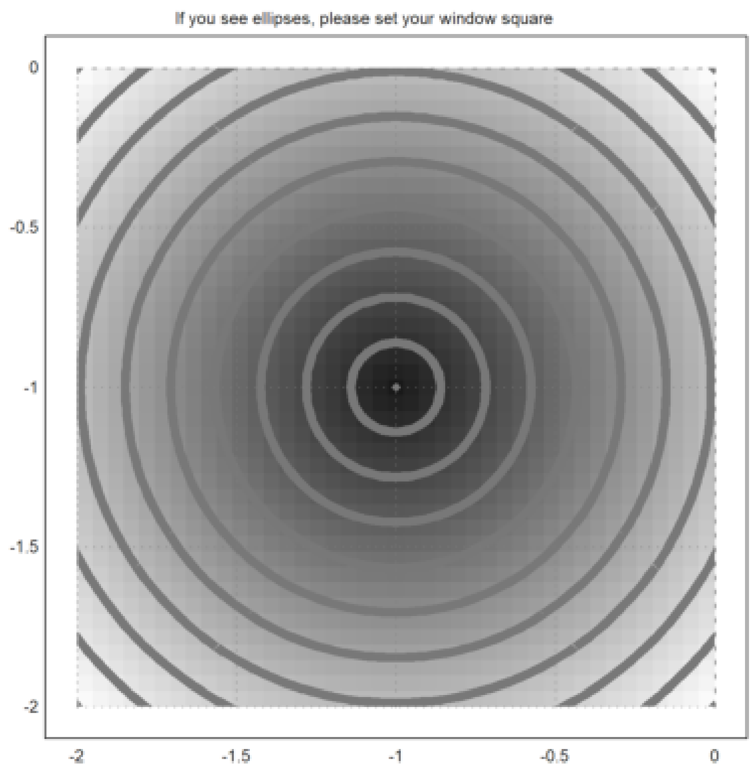
$$0$$

Contoh 6: Jarak Minimal pada Bidang

Catatan Awal

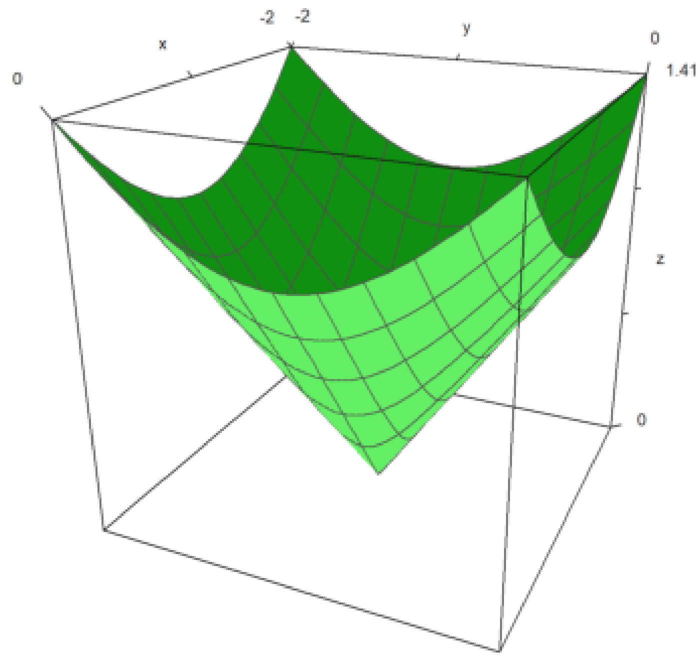
Fungsi yang, kepada sebuah titik M dalam bidang, memberikan jarak AM antara titik tetap A dan M, memiliki garis level yang cukup sederhana: lingkaran yang berpusat di A.

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



dan grafiknya juga cukup sederhana: bagian atas sebuah kerucut:

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

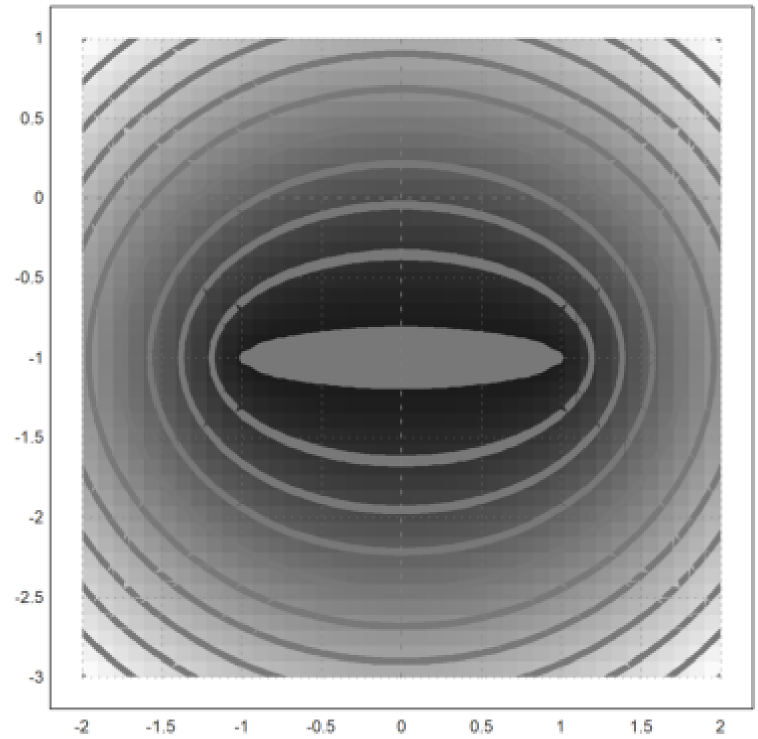


Tentu saja, minimum 0 tercapai di A.

Dua Titik

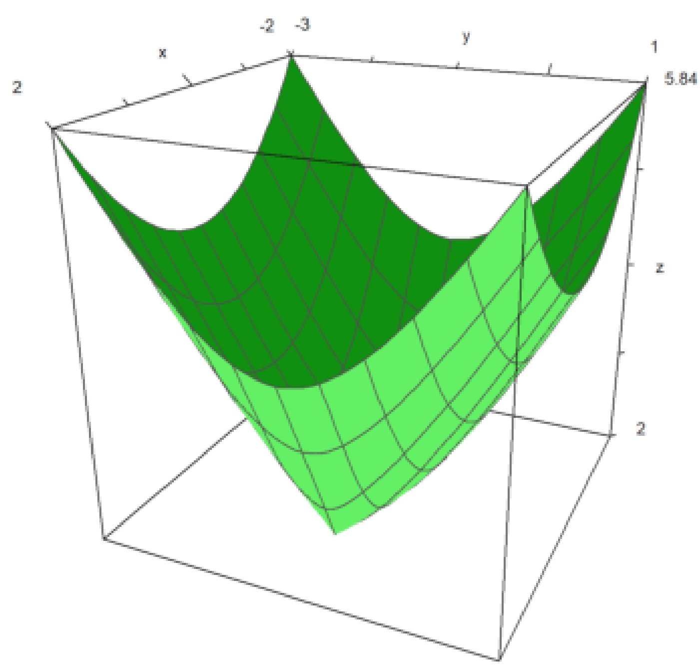
Sekarang kita melihat fungsi MA + MB di mana A dan B adalah dua titik (tetap). Ini adalah "fakta yang sudah diketahui" bahwa kurva levelnya adalah elips, dengan titik fokus adalah A dan B; kecuali untuk minimum AB yang konstan pada segmen [AB]:

```
>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):
```



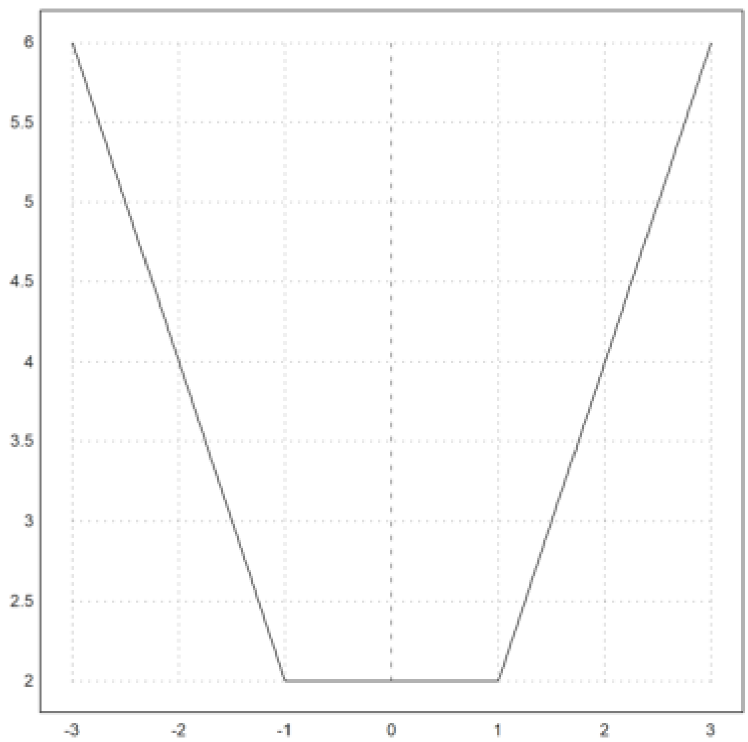
Grafiknya lebih menarik:

```
>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):
```



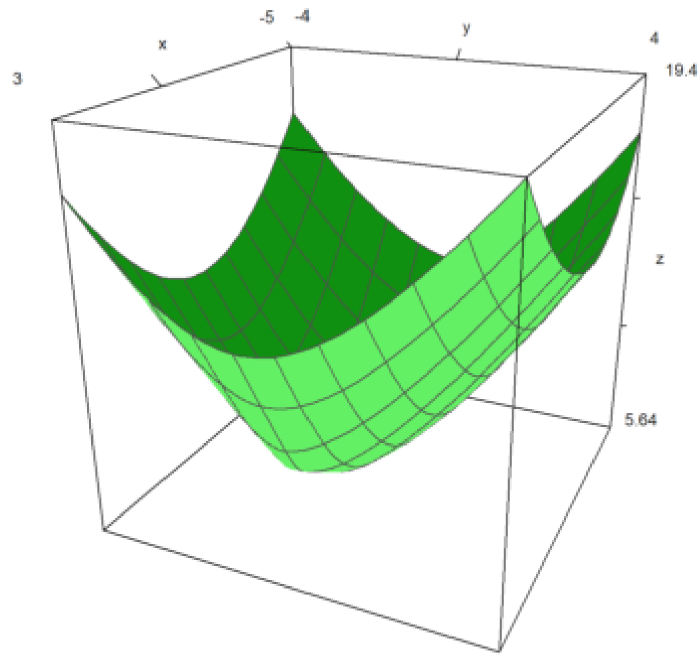
Tiga Titik

Sekarang hal-hal menjadi lebih rumit: Sedikit kurang dikenal bahwa MA + MB + MC mencapai minimumnya di satu titik di bidang, tetapi untuk menentukannya lebih sulit:

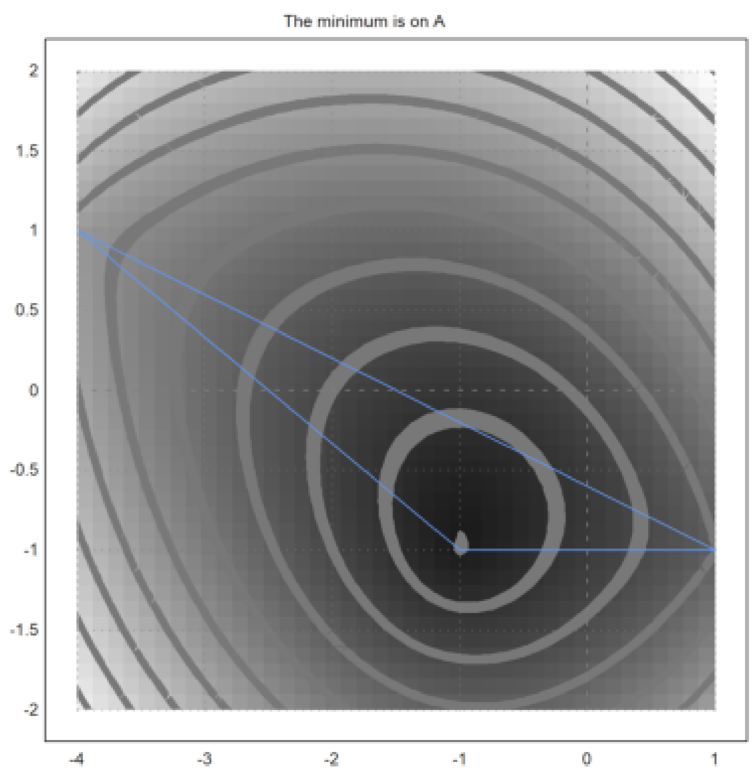
1) Jika salah satu sudut segitiga ABC lebih dari 120° (misalnya di A), maka minimumnya dicapai di titik ini (misalnya AB + AC).

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;
```

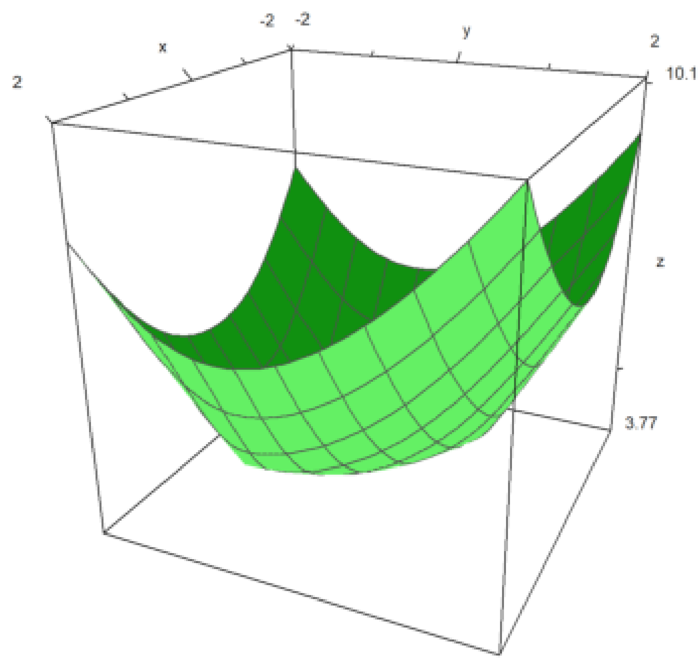



```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

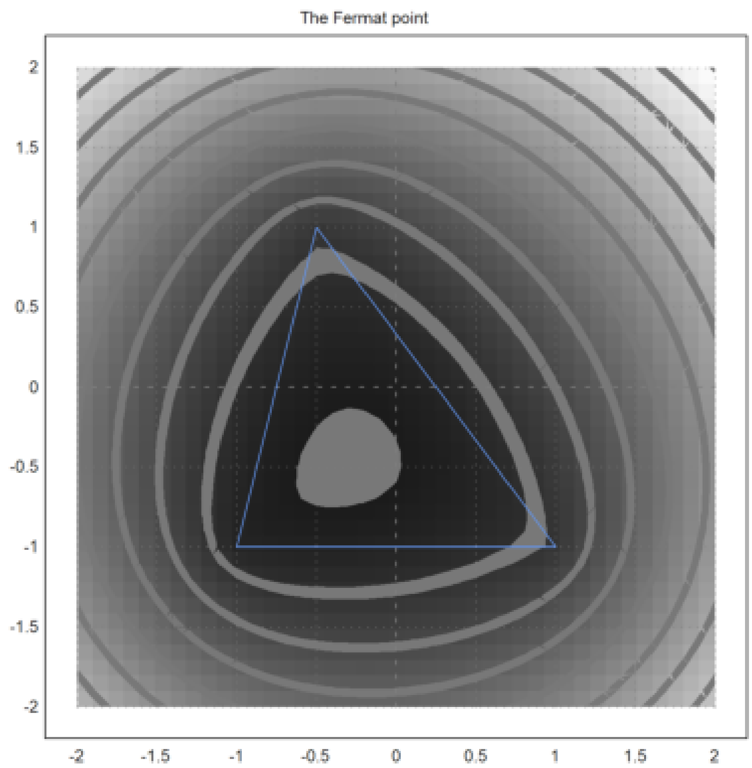


2) Tetapi jika semua sudut segitiga ABC kurang dari 120°, maka minimumnya berada di titik F di dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (masing-masing 120°):

```
>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;
```



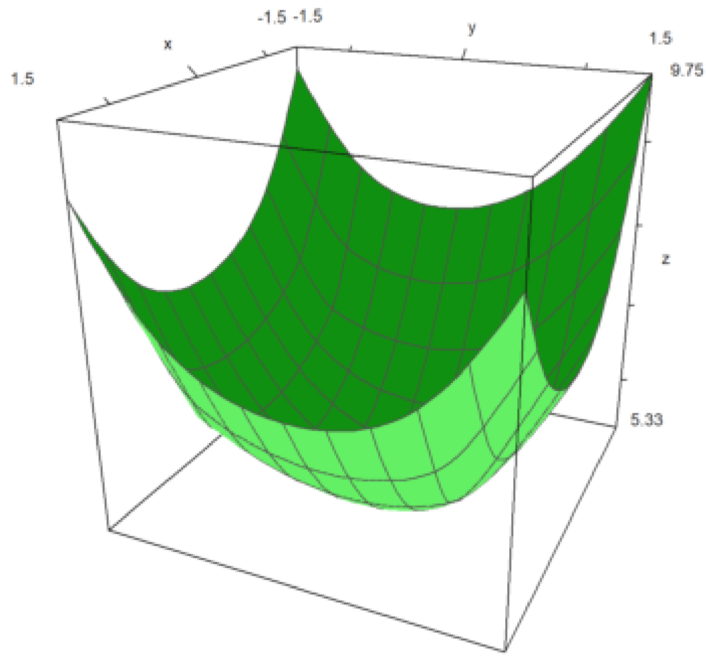
Ini adalah kegiatan menarik untuk membuat gambar di atas dengan perangkat lunak geometri; misalnya, saya tahu ada perangkat lunak yang ditulis dalam bahasa Java yang memiliki instruksi "garis kontur"...

Semua yang telah dijelaskan di atas telah ditemukan oleh seorang hakim Prancis bernama Pierre de Fermat; ia menulis surat kepada dilettan lain seperti pendeta Marin Mersenne dan Blaise Pascal yang bekerja di bidang pajak. Jadi titik tunggal F yang menjadikan FA + FB + FC minimal disebut titik Fermat dari segitiga. Tapi sepertinya beberapa tahun sebelumnya, orang Italia bernama Torriccelli telah menemukan titik ini sebelum Fermat! Namun demikian, tradisinya adalah untuk menandai titik ini sebagai F...

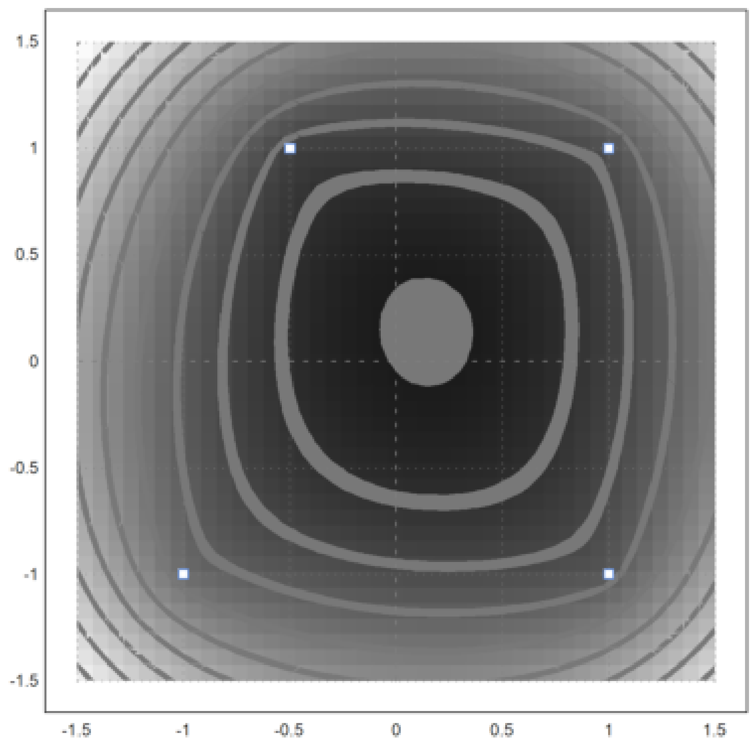
Empat Titik

Langkah berikutnya adalah menambahkan titik keempat D dan mencoba meminimalkan MA + MB + MC + MD; katakanlah Anda adalah operator TV kabel dan ingin menentukan di mana Anda harus meletakkan antena Anda sehingga Anda dapat menyediakan layanan untuk empat desa dan menggunakan sebanyak mungkin panjang kabel!

```
>D=[1,1];
>function d4(x,y):=d3(x,y)+sqrt((x-D[1])^2+(y-D[2])^2)
>plot3d("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5):
```




```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
>P=(A_B_C_D)'; plot2d(P[1],P[2],points=1,add=1,color=12);
>insimg;
```



Masih ada nilai minimum dan tidak dicapai di salah satu dari titik-titik A, B, C, atau D:

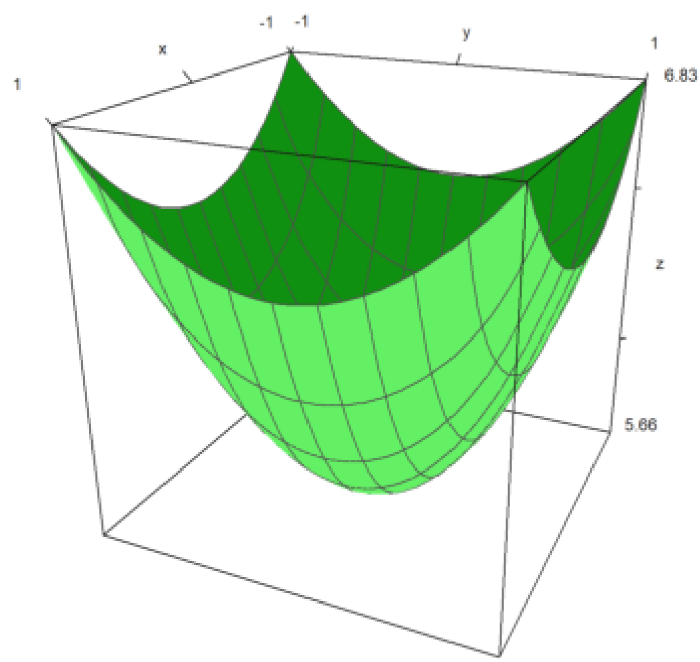
```
>function f(x):=d4(x[1],x[2])
>neldermin("f",[0.2,0.2])
```

```
[0.142858,  0.142857]
```

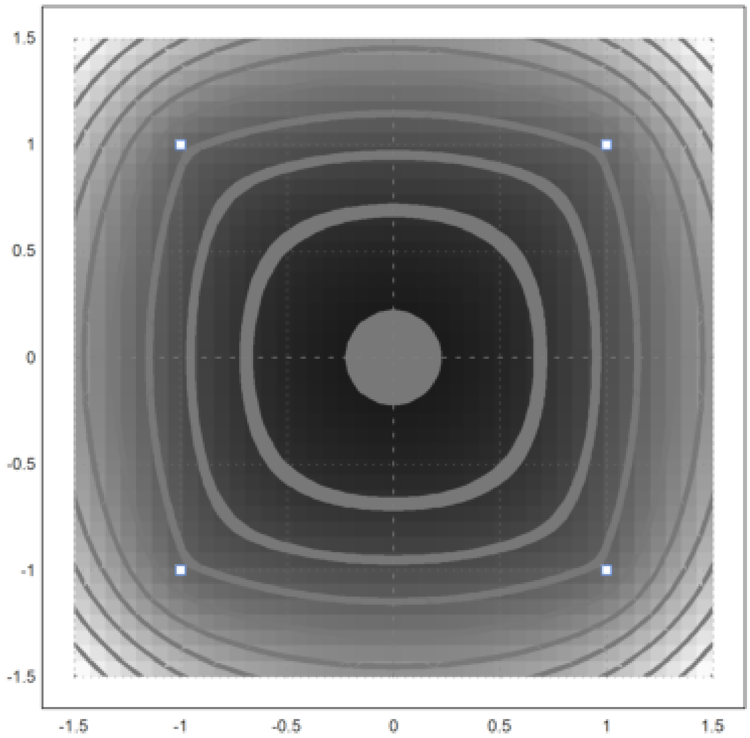
Tampaknya dalam kasus ini, koordinat titik optimal bersifat rasional atau mendekati rasional...

Sekarang ABCD adalah sebuah persegi, kita berharap bahwa titik optimal akan menjadi pusat ABCD:

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




```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
>P=(A_B_C_D)'; plot2d(P[1],P[2],add=1,color=12,points=1);
>insimg;
```



Contoh 7: Bola Dandelin dengan Povray

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

Pertama-tama kita menghitung jari-jari dari bola-bola.

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

Kami menggunakan file 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]

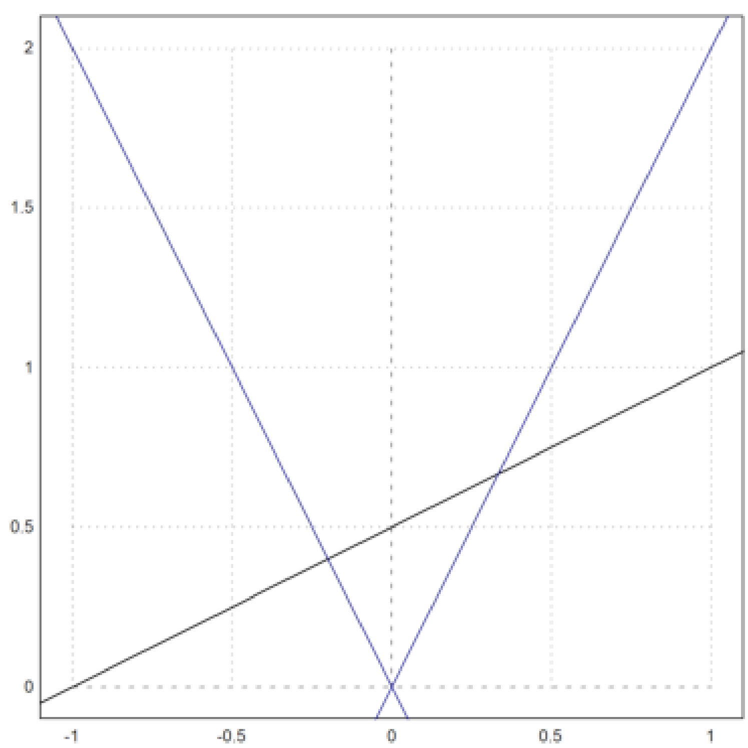
Kemudian, garis ketiga.

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

[- 1, 2, 1]

Kita menggambar semua yang telah kita buat sejauh ini.

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



Sekarang kita mengambil 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 jarak ke g.

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

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

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

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

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

Ada dua solusi.
Kami mengevaluasi solusi-solusi simbolik, dan menemukan kedua pusat, serta kedua jaraknya.

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

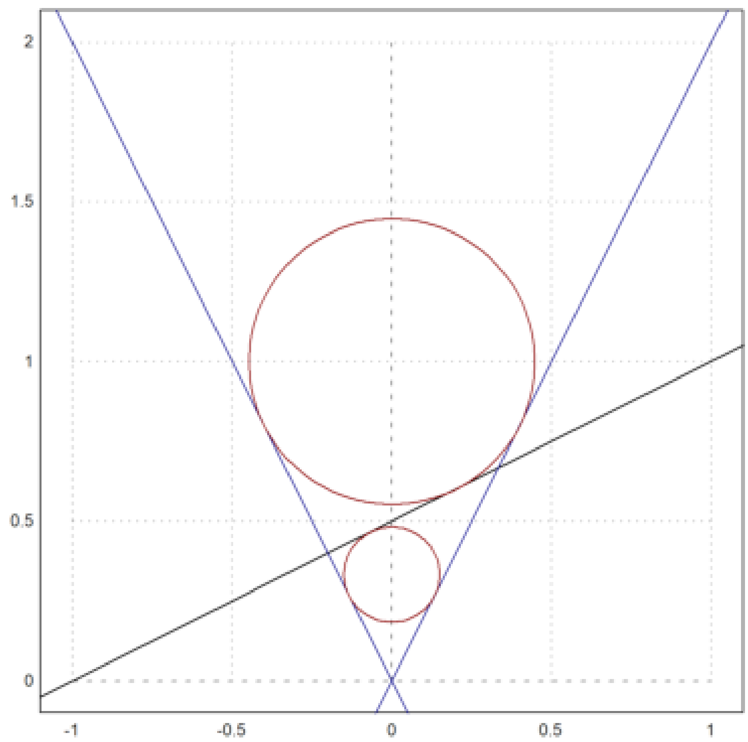
```
[0.333333,  1]
```

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

```
[0.149071,  0.447214]
```

Gambar lingkaran-lingkaran ke dalam gambar.

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



Gambar dengan Povray

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

Pertama, kita muat fungsi-fungsi Povray.

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

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

Kita atur adegan dengan tepat.

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

Selanjutnya, kita tulis dua bola ke file Povray.

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

Dan kerucut, transparan.

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

Kita menghasilkan sebuah bidang yang dibatasi oleh kerucut.

```
>gp=g();
>pc=povcone([0,0,0],0,[0,0,a],1,"");
>vp=[gp[1],0,gp[2]]; dp=gp[3];
>writeln(povplane(vp,dp,povlook(blue,0.5),pc));
```

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

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

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

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

Selanjutnya kita menghitung perpotongan antara P1 dan P2 dengan bidang.

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

Kita menghubungkan titik-titik tersebut dengan segmen garis.

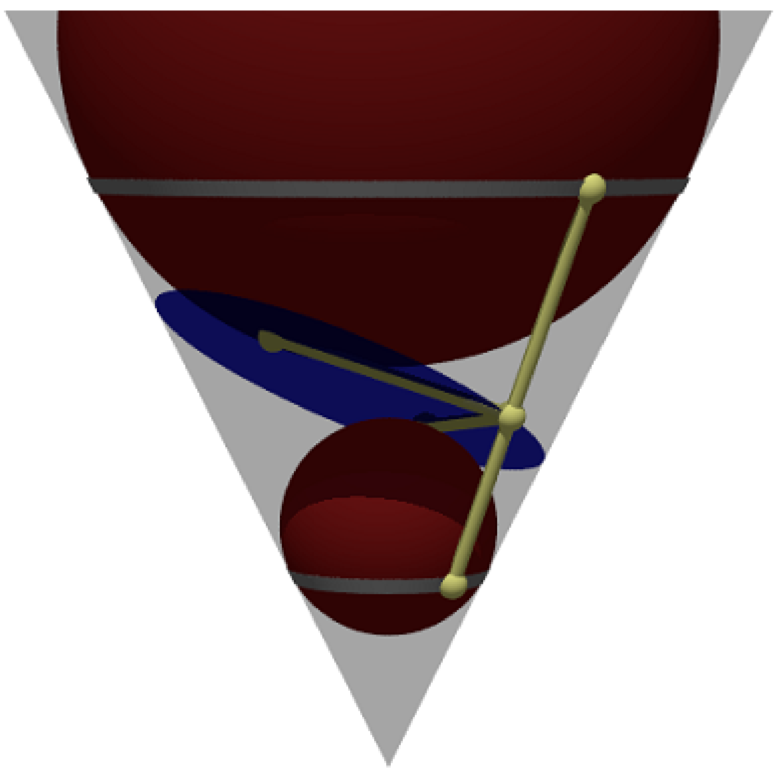
```
>writeln(povsegment(P1,P2,povlook(yellow)));
>writeln(povsegment(P5,P3,povlook(yellow)));
>writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang kita menghasilkan sebuah band abu-abu di mana bola menyentuh kerucut.

```
>pcw=povcone([0,0,0],0,[0,0,a],1.01);
>pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize
>writeln(povintersection([pcw,pc1],povlook(gray)));
>pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsize
>writeln(povintersection([pcw,pc2],povlook(gray)));
```

Mulai program Povray.

```
>povend();
```



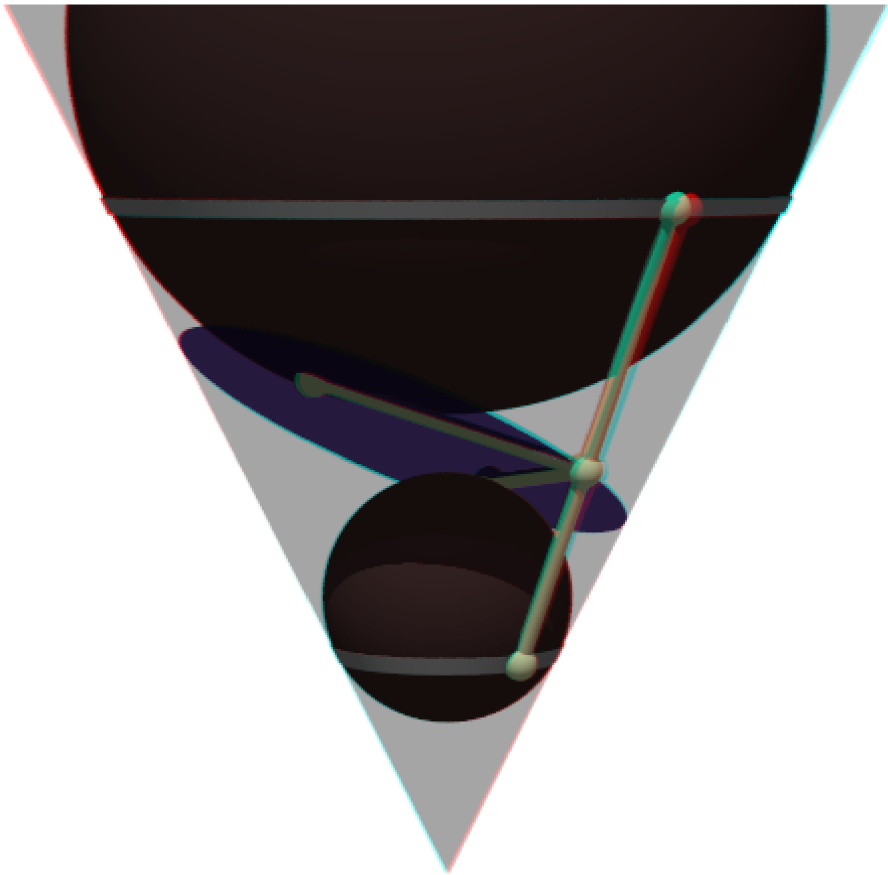
Untuk mendapatkan Anaglyph dari ini, kita perlu memasukkan semuanya ke dalam fungsi adegan. Fungsi ini akan digunakan dua kali nanti.

```
>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);
  pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsiz
  writeln(povintersection([pcw,pc1],povlook(gray)));
  pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsiz
  writeln(povintersection([pcw,pc2],povlook(gray)));
  endfunction
```

Anda memerlukan kacamata merah/biru untuk menghargai efek berikut.

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



Contoh 8: Geometri Bumi

Dalam catatan ini, kita ingin melakukan beberapa perhitungan bola. Fungsi-fungsi tersebut 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 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‘ (pencetakan posisi bola).

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

```
S 7°46.467' E 110°23.050'
```

Mari tambahkan dua kota lagi, Solo dan Semarang.

```
>Solo=[rad(-7,-34.333),rad(110,49.683)]; Semarang=[rad(-6,-59.05),rad(110,2
>sposprint(Solo), sposprint(Semarang),
```

```
S 7°34.333' E 110°49.683'
S 6°59.050' E 110°24.533'
```

Pertama-tama kita menghitung vektor dari satu kota ke kota lainnya pada bola ideal. Vektor ini adalah [heading, jarak] dalam radian. Untuk menghitung jarak di bumi, kita mengalikannya dengan radius bumi pada lintang 7°.

```
>br=svector(FMIPA,Solo); degprint(br[1]), br[2]*rearth(7°)->km // perkiraan
```

```
65°20'26.60''
53.8945384608
```

Ini adalah perkiraan yang baik. Rutinitas berikut menggunakan perkiraan yang lebih baik. Pada jarak yang begitu pendek, hasilnya hampir sama.

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

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

Terdapat fungsi untuk perhitungan heading, dengan mempertimbangkan bentuk elips bumi. Sekali lagi, kita mencetaknya dengan cara yang lebih canggih.

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

```
     65.34°
```

Sudut suatu segitiga melebihi 180° pada bola.

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

```
180°0'10.77''
```

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

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

```
Commands must be separated by semicolon or comma!
Found:  // perkiraan luas segitiga FMIPA-Solo-Semarang (character 32)
You can disable this in the Options menu.
Error in:
(asum-pi)*rearth(48°)^2->" km^2" // perkiraan luas segitiga FM ...
                                ^
```

There is a function for this, which uses the mean latitude of the triangle to compute the earth radius, and takes care of rounding errors for very small triangles.

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

```
2123.64310526 km^2
```

Kita juga dapat menambahkan vektor ke posisi. Sebuah vektor berisi heading 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 untuk bumi.

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

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

Mari kita beralih ke contoh yang lebih besar, Tugu Jogja dan Monas Jakarta (menggunakan Google Earth untuk mencari koordinatnya).

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

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

Menurut Google Earth, jaraknya adalah 429.66km. Kami mendapatkan perkiraan yang baik.

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

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

Heading-nya sama dengan yang dihitung di Google Earth.

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

```
294°17'2.85''
```

Namun, kita tidak lagi mendapatkan posisi target yang tepat jika kita menambahkan heading dan jarak ke posisi awal. Ini terjadi karena kita tidak menghitung fungsi invers secara tepat, tetapi mengambil perkiraan radius bumi sepanjang jalur.

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

```
S 6°10.500' E 106°48.717'
```

Kesalahan ini tidak besar, namun demikian.

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

```
S 6°10.500' E 106°48.717'
```

Tentu saja, kita tidak dapat berlayar dengan heading yang sama dari satu tujuan ke tujuan lainnya jika kita ingin mengambil jalur terpendek. Bayangkan Anda terbang ke arah timur laut (NE) dimulai dari titik manapun di bumi. Maka Anda akan bergerak dalam spiral menuju kutub utara. Lingkaran besar tidak mengikuti heading yang konstan!

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

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

Sekarang kita menambahkan 10 kali satu per sepuluh dari jarak, menggunakan heading ke Monas yang kita dapatkan di Tugu.

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

Hasilnya jauh dari yang benar.

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

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

Sebagai contoh lain, mari 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 utara dari P1.

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

```
     79.69°
```

Namun, jika kita mengikuti arah mata angin ini, kita akan berputar ke kutub utara! Jadi kita harus menyesuaikan heading kita sepanjang perjalanan. Untuk tujuan 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); e
```

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

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

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

```
0.203km
```

Kita mendapatkan perkiraan yang baik jika kita menyesuaikan heading kita setiap 1/100 dari jarak total 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 tujuan navigasi, kita dapat mendapatkan rangkaian posisi GPS di sepanjang lingkaran besar ke 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'
```

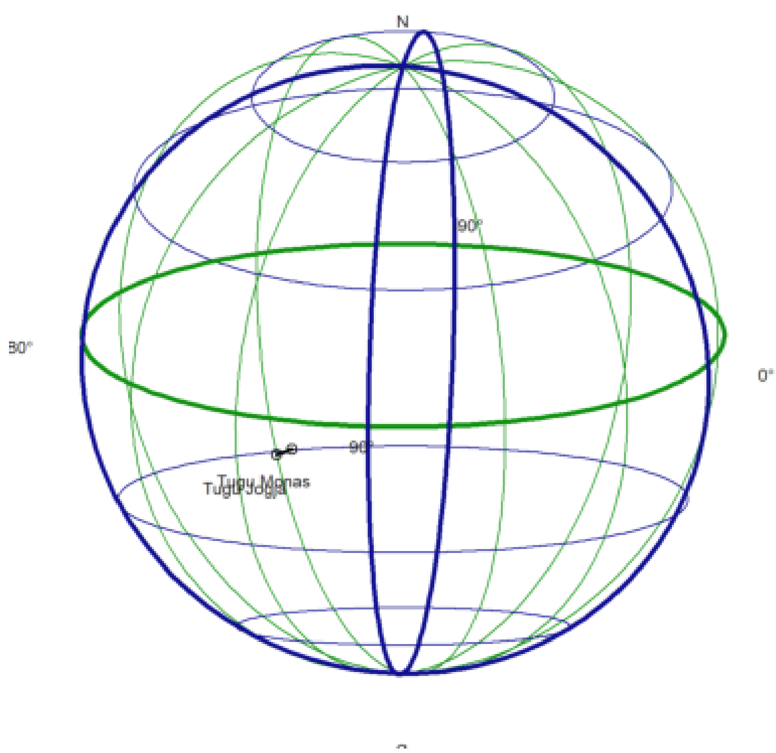
Kita menulis sebuah fungsi yang menggambar bumi, dua posisi, dan posisi di antaranya.

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

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

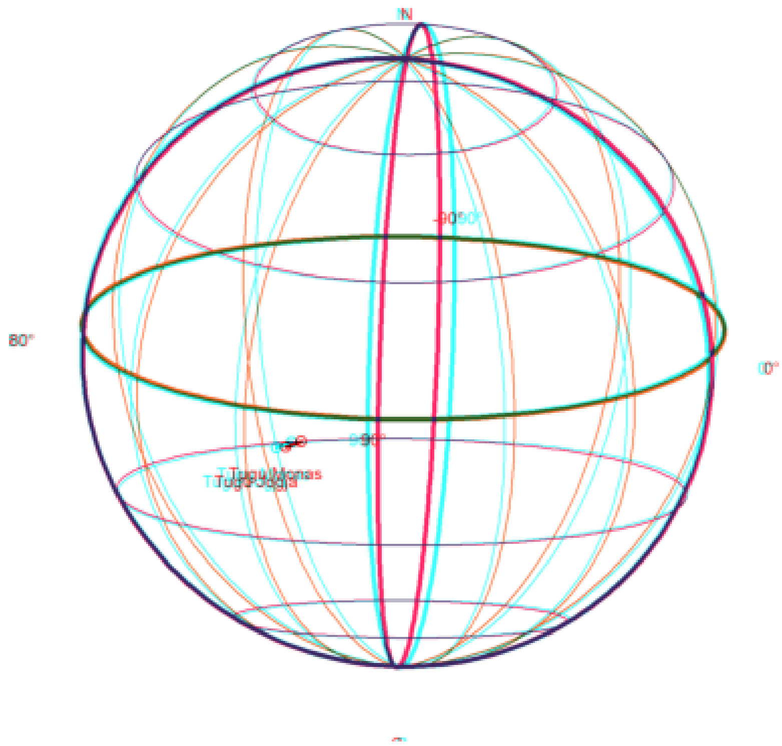
Sekarang gambar semuanya.

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



Atau gunakan 'plot3d' untuk mendapatkan tampilan anaglyph. Ini terlihat sangat bagus dengan kacamata merah/biru.

```
>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 kali panjang sisi-sisi yang berhadapan sama.

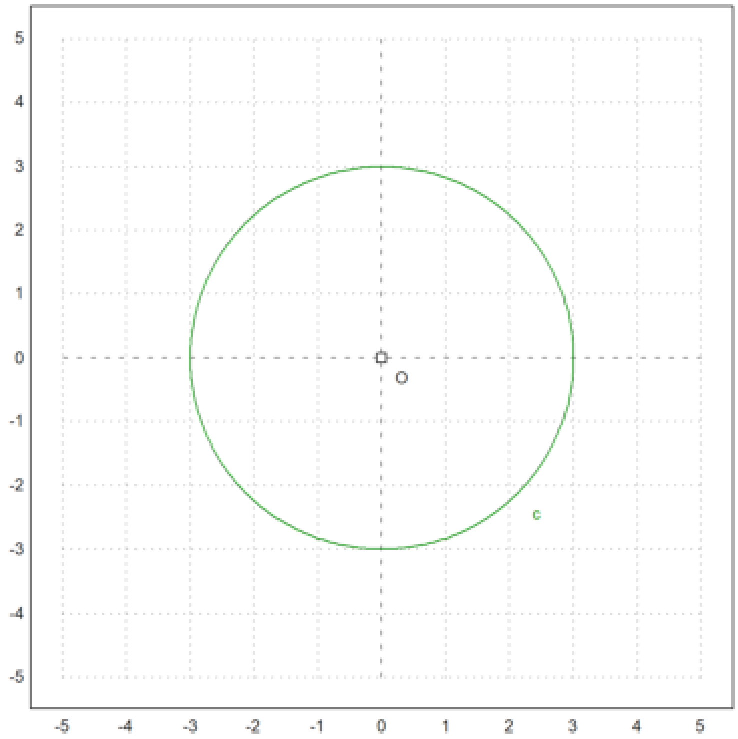
4. Gambarlah suatu ellips jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang jumlah jarak ke P dan ke Q selalu sama (konstan).

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).

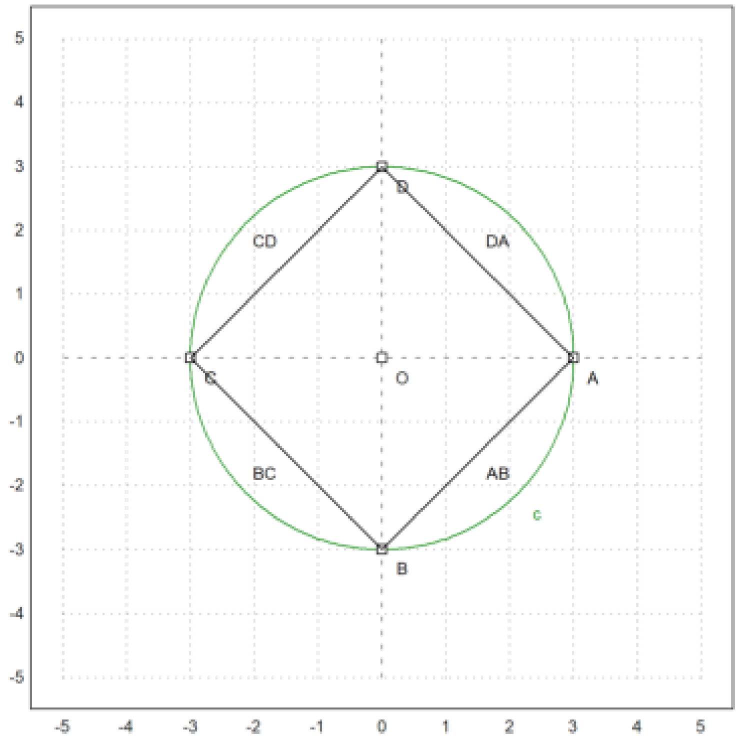
Penyelesaian

1. Akan digambar 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.

```
>setPlotRange(5);
>O=[0,0]; plotPoint(O,"O");
>r=3;
>color(3); plotCircle(circleWithCenter(O,r)): //gambar lingkaran
```

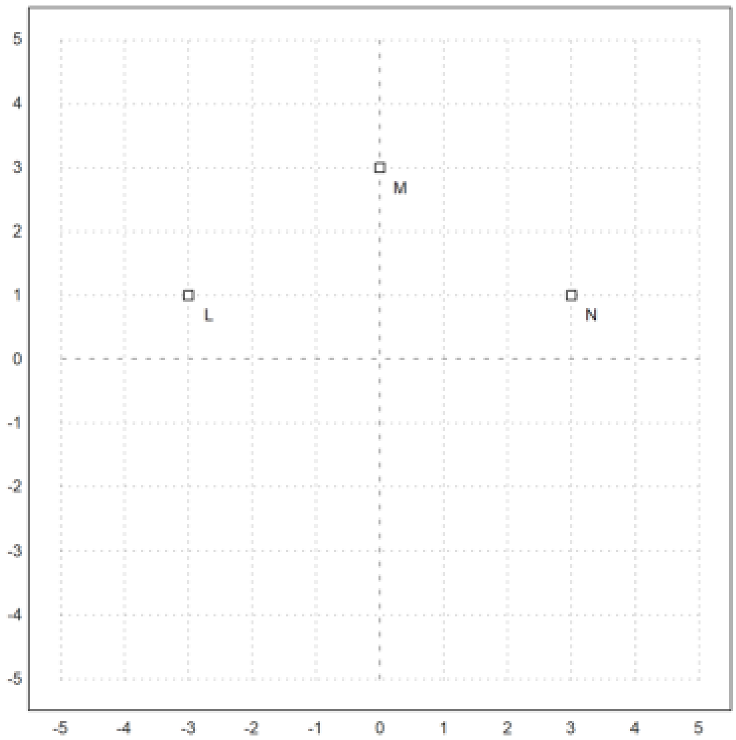



```
>A=[3,0]; B=[0,-3]; C=[-3,0]; D=[0,3];
>color(1); plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); plotPoint(
>plotSegment(A,B,"AB"); plotSegment(B,C,"BC"); plotSegment(C,D,"CD"); plotS
```



2. Akan digambar suatu parabola yang melalui 3 titik yang diketahui.

```
>setPlotRange(5);
>L=[-3,1]; N=[3,1]; M=[0,3];
>color(1); plotPoint(L,"L"); plotPoint(N,"N"); plotPoint(M,"M"):
```



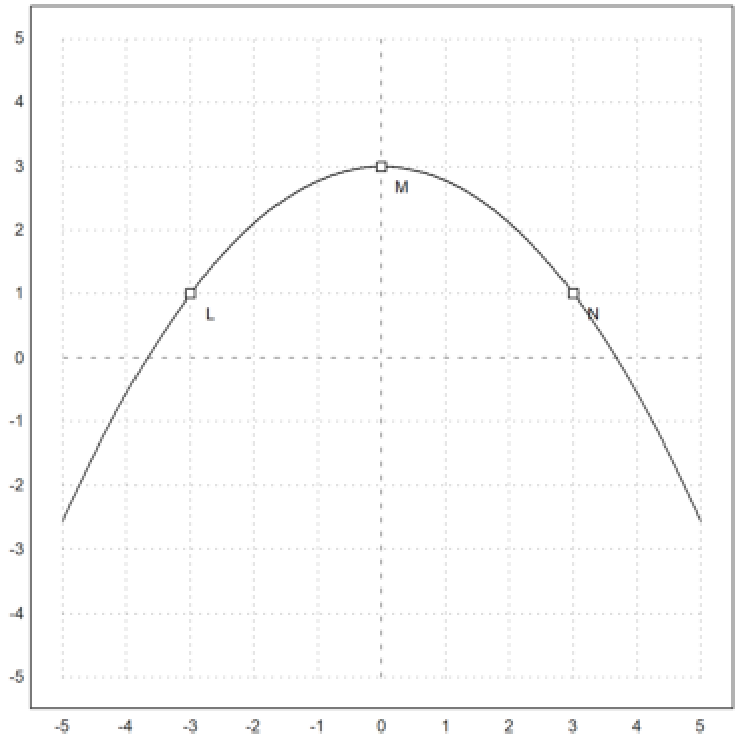

```
>sol &= solve([9*a-3*b+c=1,9*a+3*b+c=1,c=3],[a,b,c])
```

```
                 2
        [[a = - -, b = 0, c = 3]]
                 9
```

```
>function y(x)&=(-2/9)*(x^2)-0*x+3
```

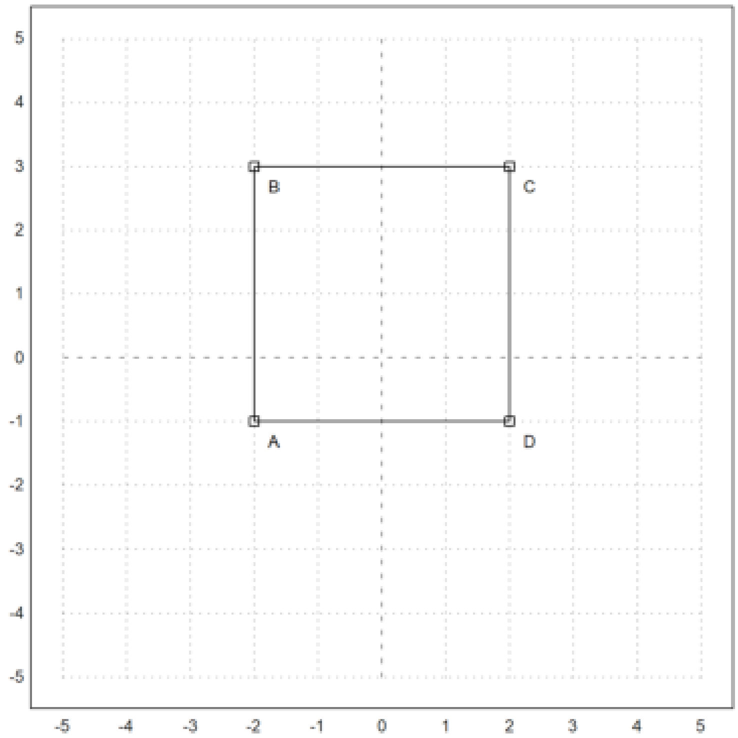
```
                        2
                     2 x
                 3 - ----
                      9
```

```
>plot2d(y,r=5); color(1); plotPoint(L,"L"); plotPoint(N,"N"); plotPoint(M,"
```



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

```
>setPlotRange(5);
>A=[-2,-1]; B=[-2,3]; C=[2,3]; D=[2,-1];
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); plotPoint(D,"D");
>plotSegment(A,B,""); plotSegment(B,C,""); plotSegment(C,D,""); plotSegment
```




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

```
[0,  1]
```

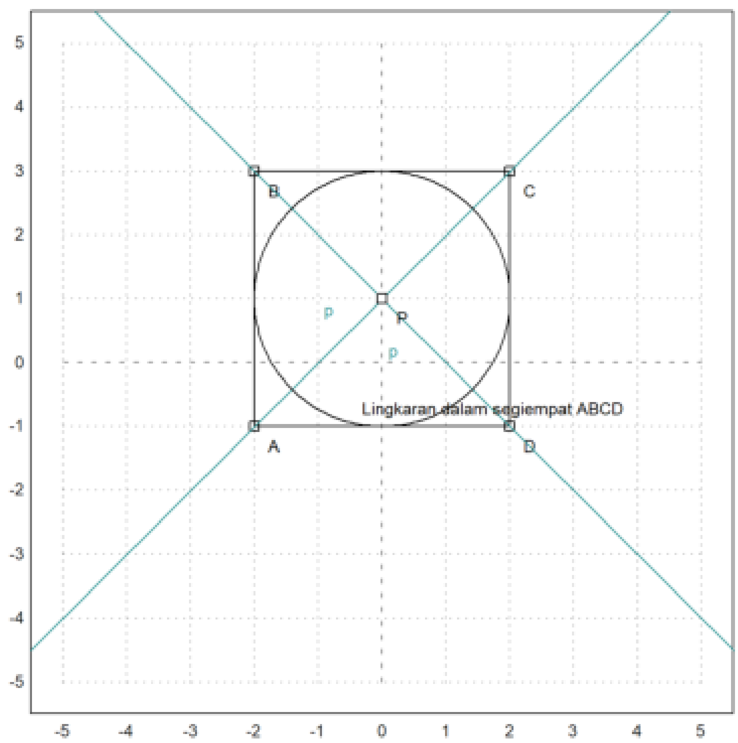
```
>color(5); plotLine(k); plotLine(l); color(3); // gambar kedua garis bagi s
>color(1); plotPoint(P,"P"); // gambar titik potongnya
```

Segiempat tersebut merupakan segiempat garis singgung karena keempat garis bagi sudutnya bertemu di satu titik. Selanjutnya, akan digambar lingkaran dalamnya.

```
>r=norm(P-projectToLine(P,lineThrough(A,B))) // jari-jari lingkaran dalam
```

```
2
```

```
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segiempat ABCD"): // gam
```



Selanjutnya akan ditunjukkan bahwa syarat suatu segiempat merupakan segiempat garis singgung apabila hasil kali panjang sisi-sisi yang berhadapan sama.

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

4

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

4

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

4

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

4

```
>AB*CD
```

```
16
```

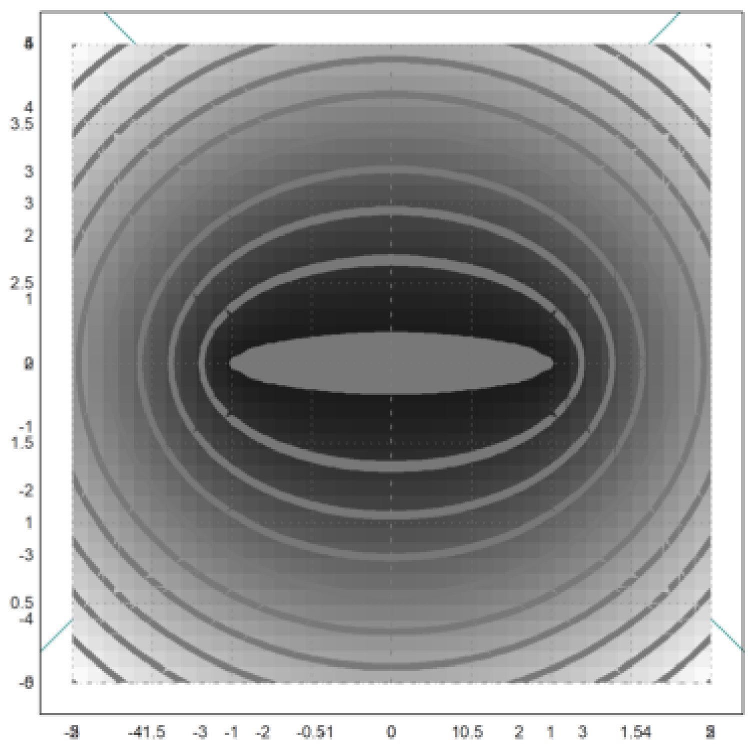
```
>AD*BC
```

```
16
```

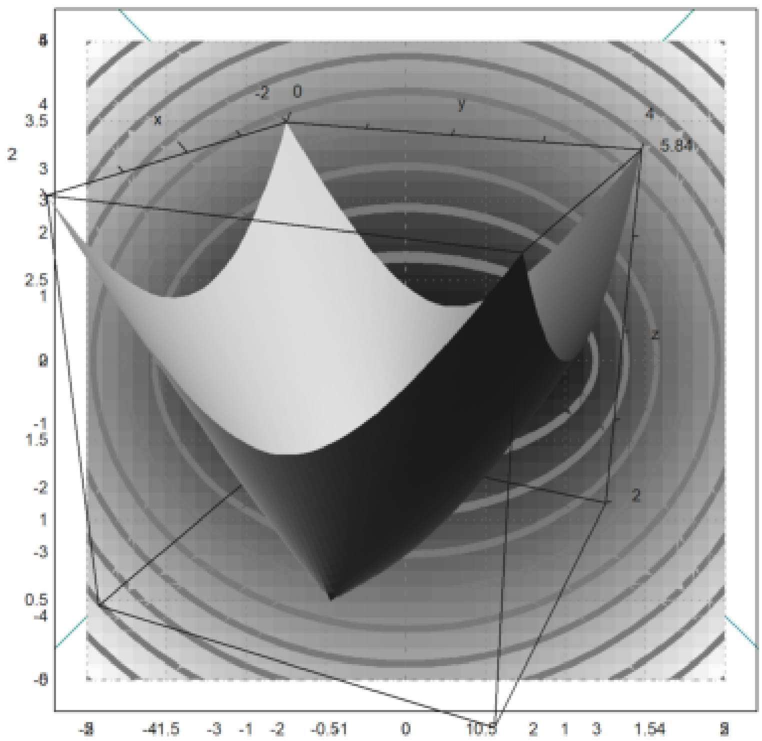
Karena hasil kali AB x CD sama dengan hasil kali AD x BC, maka segiempat tersebut merupakan segiempat garis singgung.

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

```
>P=[-1,2]; Q=[1,2];
>function d1(x,y):=sqrt((x-P[1])^2+(y-P[2])^2)
>function d2(x,y):=d1(x,y)+sqrt((x-Q[1])^2+(y-Q[2])^2)
>fcontour("d2",xmin=-2,xmax=2,ymin=0,ymax=4,hue=1):
```

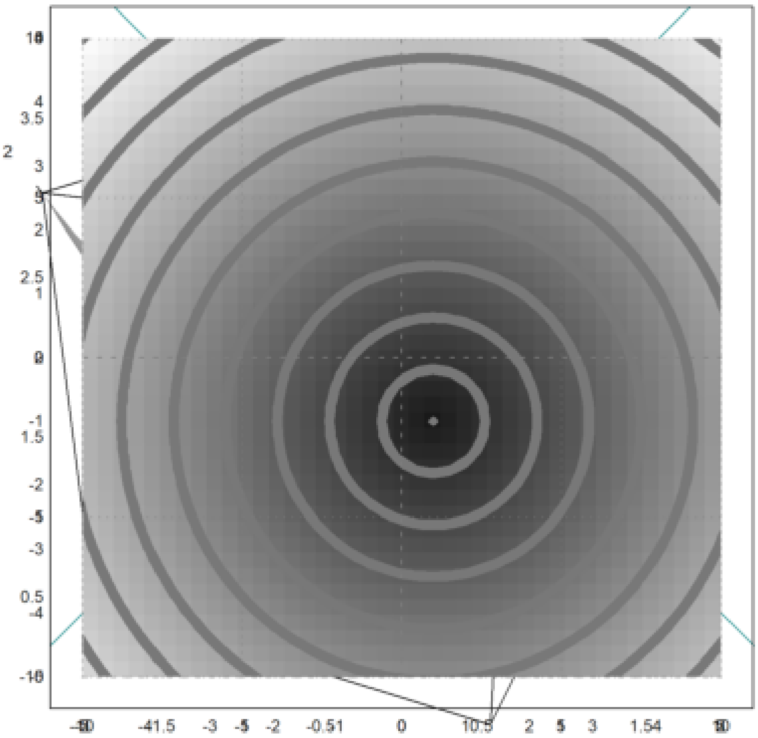


```
>plot3d("d2",xmin=-2,xmax=2,ymin=0,ymax=4,hue=1):
```

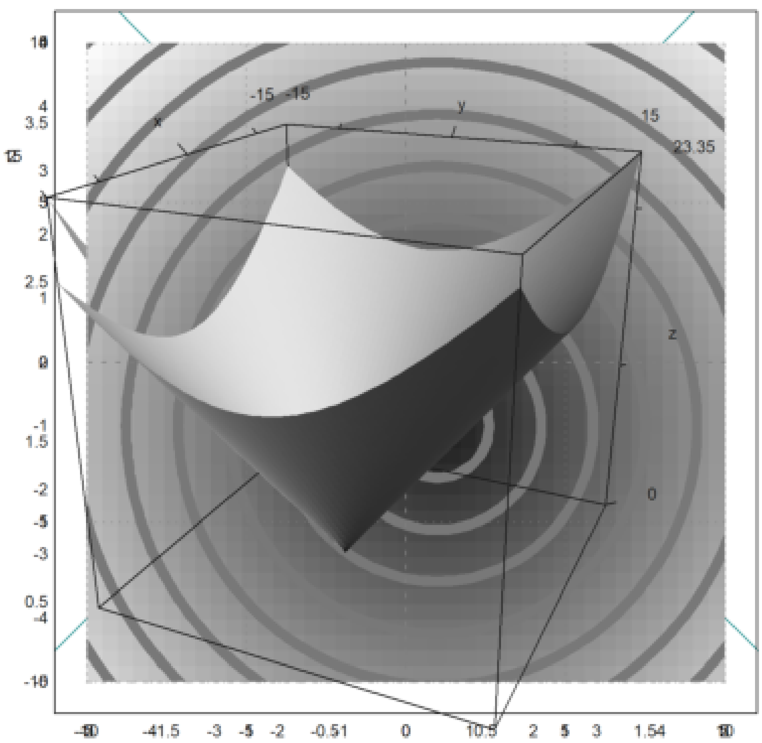


5. Akan digambar suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).

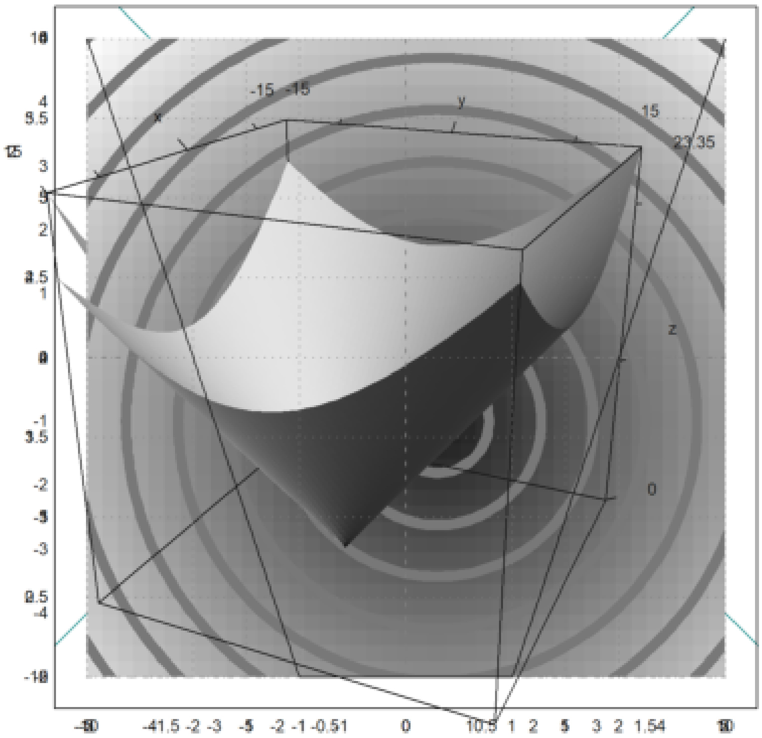
```
>P=[-2,1]; Q=[2,1];
>function d3(x,y):=sqrt((x-P[2])^2+(y-P[1])^2)
>function d4(x,y):=d3(x,y)+sqrt((x-Q[2])^2+(y-Q[1])^2)
>fcontour("d3",r=10,hue=1):
```



```
>plot3d("d3",r=15,hue=1):
```



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



BAB 7

STATISTIKA DENGAN EMT

EMT untuk Statistika

Dalam buku catatan ini, kami menunjukkan plot statistik utama, uji, dan distribusi dalam Euler.

Mari kita mulai dengan beberapa statistik deskriptif. Ini bukan pengantar statistika. Jadi, Anda mungkin memerlukan beberapa pengetahuan dasar untuk memahami rincian tersebut.

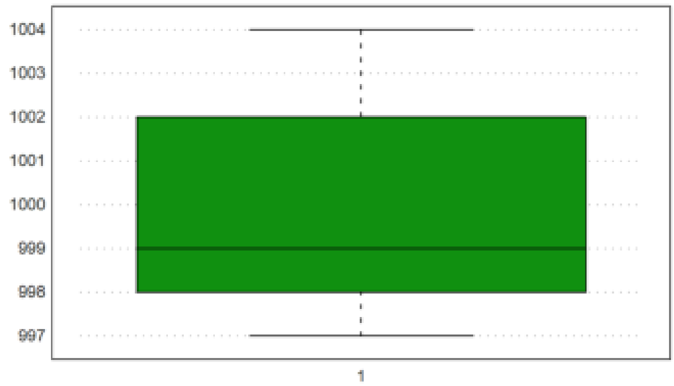
Anggaplah pengukuran berikut. Kami ingin menghitung nilai rata-rata dan deviasi standar yang diukur.

```
>M=[1000,1004,998,997,1002,1001,998,1004,998,997]; ...
>median(M), mean(M), dev(M),
```

```
999
999.9
2.72641400622
```

Kita dapat membuat plot diagram kotak dan garis (box-and-whiskers) untuk data ini. Dalam kasus kita, tidak ada data yang berada di luar jangkauan (outliers).

```
>aspect(1.75); boxplot(M):
```



Kami menghitung probabilitas bahwa suatu nilai lebih besar dari 1005, dengan asumsi nilai yang diukur berasal dari distribusi normal.

Semua fungsi distribusi dalam Euler diakhiri dengan ...dis dan menghitung distribusi probabilitas kumulatif (CPD).

$$\text{normaldis(x,m,d)} = \int_{-\infty}^{x} \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t-m}{d})^2}\ dt.$$

Kami mencetak hasilnya dalam bentuk persen dengan akurasi dua digit desimal menggunakan fungsi print.

```
>print((1-normaldis(1005,mean(M),dev(M)))*100,2,unit=" %")
```

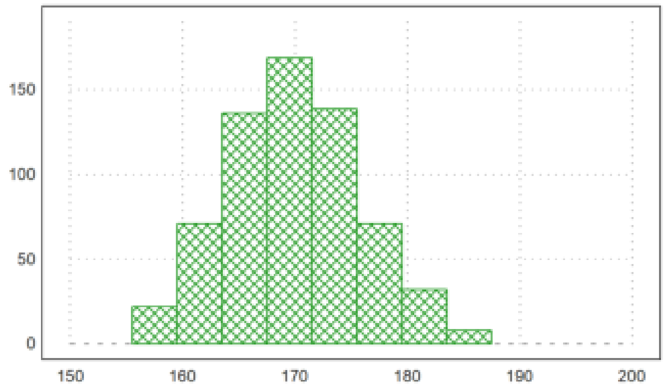
```
3.07 %
```

Untuk contoh berikutnya, kami mengasumsikan jumlah pria dalam rentang ukuran yang diberikan sebagai berikut.

```
>r=155.5:4:187.5; v=[22,71,136,169,139,71,32,8];
```

Berikut adalah plot dari distribusinya.

```
>plot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style="\/"):
```



Kita dapat memasukkan data mentah seperti ini ke dalam tabel.

Tabel adalah metode untuk menyimpan data statistik. Tabel kita seharusnya memiliki tiga kolom: Awal rentang, akhir rentang, jumlah pria dalam rentang tersebut.

Tabel dapat dicetak dengan judul. Kami menggunakan vektor string untuk menetapkan judul kolom.

```
>T:=r[1:8]' | r[2:9]' | v'; writetable(T,labc=["BB","BA","Frek"])
```

```
       BB        BA      Frek
    155.5     159.5        22
    159.5     163.5        71
    163.5     167.5       136
    167.5     171.5       169
    171.5     175.5       139
    175.5     179.5        71
    179.5     183.5        32
    183.5     187.5         8
```

Jika kita membutuhkan nilai rata-rata dan statistik lainnya dari ukuran tersebut, kita perlu menghitung titik tengah dari rentang tersebut. Kita dapat menggunakan dua kolom pertama dari tabel kita untuk ini.

Simbol "|" digunakan untuk memisahkan kolom, fungsi "writetable" digunakan untuk menulis tabel, dengan opsi "labc" digunakan untuk menentukan judul kolom.

```
>(T[,1]+T[,2])/2 // nilai tengah dari tiap interval
```

```
    157.5
    161.5
    165.5
    169.5
    173.5
```

```
177.5
181.5
185.5
```

Namun, lebih mudah untuk menjumlahkan rentang tersebut dengan vektor [1/2, 1/2].

```
>M=fold(r,[0.5,0.5])
```

```
[157.5,  161.5,  165.5,  169.5,  173.5,  177.5,  181.5,  185.5]
```

Sekarang kita dapat menghitung nilai rata-rata dan deviasi dari sampel dengan frekuensi yang diberikan.

```
>{m,d}=meandev(M,v); m, d,
```

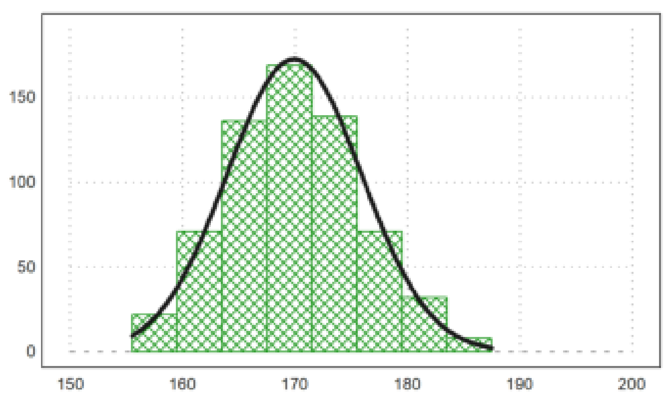
```
169.901234568
5.98912964449
```

Mari tambahkan distribusi normal dari nilai-nilai tersebut ke plot batang di atas. Rumus untuk distribusi normal dengan rata-rata m dan deviasi standar d adalah:

$$y = \frac{1}{d\sqrt{2\pi}} e^{\frac{-(x-m)^2}{2d^2}}.$$

Karena nilai-nilainya berada di antara 0 dan 1, untuk memplotnya pada diagram batang, nilai tersebut harus dikalikan dengan 4 kali jumlah total data.

```
>plot2d("qnormal(x,m,d)*sum(v)*4", ...
>  xmin=min(r),xmax=max(r),thickness=3,add=1):
```



Contoh Soal

1) Tentukan median, mean, deviasi, dan distribusi kumulatif normal (CDF) dari data tersebut.

$$N = [199, 200, 201, 202, 203, 204, 205, 205, 206, 207, 208, 208, 209]$$

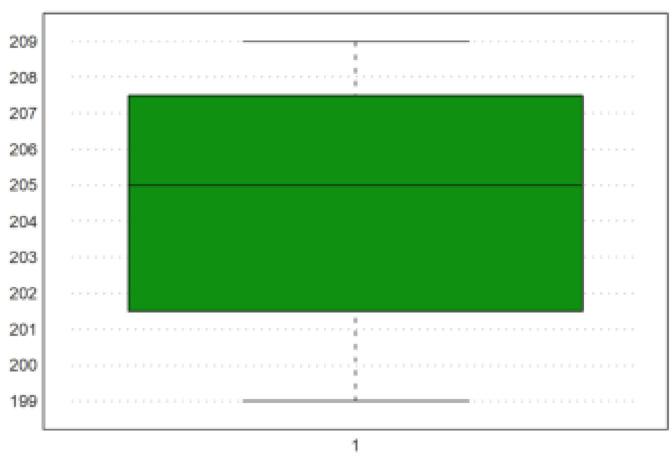
```
>N=[199, 200, 201,202,203,204,205,205,206,207,208,208,209]
```

```
[199,  200,  201,  202,  203,  204,  205,  205,  206,  207,  208,  208,
209]
```

```
>median(N), mean(N), dev(N),
```

```
205
204.384615385
3.22847904176
```

```
>aspect(1.5); boxplot(N):
```

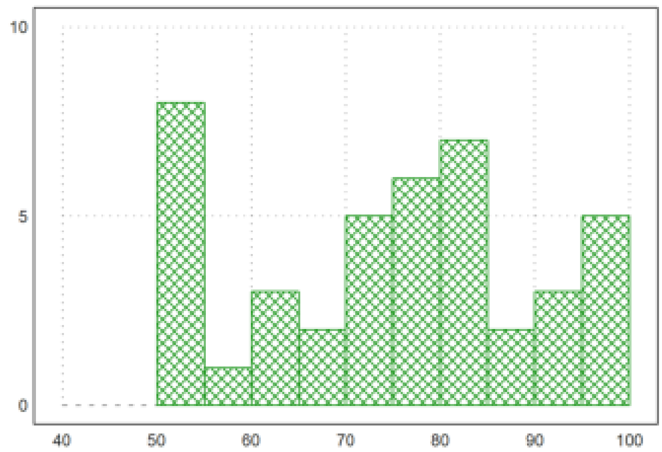



```
>print((1-normaldis(199,mean(N),dev(N)))*100,2,unit=" %")
```

```
95.23 %
```

2) Buatlah plot dari jumlah siswa dalam rentang yang diberikan sebagai berikut.

```
>r=50:5:100; v=[8,1,3,2,5,6,7,2,3,5];
>plot2d(r,v,a=40,b=100,c=0,d=10,bar=1,style="\/"):
```



Tabel

Di direktori buku catatan ini, Anda akan menemukan sebuah file berisi tabel. Data tersebut mewakili hasil dari sebuah survei. Berikut adalah empat baris pertama dari file tersebut. Data ini berasal dari sebuah buku online Jerman "Einführung in die Statistik mit R" oleh A. Handl.

```
>printfile("table.dat",4); //function printfile (filename, lines)
```

```
Could not open the file
table.dat
for reading!
Try "trace errors" to inspect local variables after errors.
printfile:
    open(filename,"r");
```

Tabel tersebut berisi 7 kolom angka atau token (string). Kami ingin membaca tabel dari file tersebut. Pertama, kami akan menggunakan terjemahan kami sendiri untuk token-token tersebut.

Untuk melakukan ini, kami mendefinisikan set token-token tersebut. Fungsi strtokens() mendapatkan vektor string token dari sebuah string yang diberikan.

```
>mf:=["m","f"]; yn:=["y","n"]; ev:=strtokens("g vg m b vb");
```

Sekarang kita akan membaca tabel dengan terjemahan ini.

Argumen tok2, tok4, dan sebagainya adalah terjemahan dari kolom-kolom tabel. Argumen-argumen ini tidak ada dalam daftar parameter readtable(), sehingga Anda perlu menyediakannya dengan ":=".

```
>{MT,hd}=readtable("table.dat",tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
```

```
Could not open the file
table.dat
for reading!
Try "trace errors" to inspect local variables after errors.
readtable:
    if filename!=none then open(filename,"r"); endif;
```

```
>load over statistics;
```

Untuk mencetak, kita perlu menentukan set token yang sama. Kita hanya akan mencetak empat baris pertama.

```
>writetable(MT[1:10],labc=hd,wc=5,tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
```

```
MT is not a variable!
Error in:
writetable(MT[1:10],labc=hd,wc=5,tok2:=mf,tok4:=yn,tok5:=ev,to ...
                   ^
```

Titik "." mewakili nilai yang tidak tersedia.

Jika kita tidak ingin menentukan token untuk terjemahan sebelumnya, kita hanya perlu menentukan kolom-kolom yang berisi token, bukan angka..

```
>ctok=[2,4,5,7]; {MT,hd,tok}=readtable("table.dat",ctok=ctok);
```

```
Could not open the file
table.dat
for reading!
Try "trace errors" to inspect local variables after errors.
readtable:
    if filename!=none then open(filename,"r"); endif;
```

Fungsi readtable() sekarang mengembalikan sebuah set token.

```
>tok
```

```
Variable tok not found!
Error in:
tok ...
   ^
```

Tabel tersebut berisi entri dari file dengan token diterjemahkan menjadi angka.

String khusus NA="." diartikan sebagai "Not Available" (Tidak Tersedia), dan diubah menjadi NAN (bukan angka) dalam tabel. Terjemahan ini dapat diubah dengan parameter NA dan NAval.

```
>MT[1]
```

```
MT is not a variable!
Error in:
MT[1] ...
     ^
```

Berikut adalah konten tabel dengan angka yang tidak diterjemahkan.

```
>writetable(MT,wc=5)
```

```
Variable or function MT not found.
Error in:
writetable(MT,wc=5) ...
             ^
```

Untuk kenyamanan, Anda dapat menyimpan output dari readtable() ke dalam sebuah daftar (list).

```
>Table={{readtable("table.dat",ctok=ctok)}};
```

```
Could not open the file
table.dat
for reading!
Try "trace errors" to inspect local variables after errors.
readtable:
    if filename!=none then open(filename,"r"); endif;
```

Dengan menggunakan kolom-kolom token yang sama dan token yang dibaca dari file, kita dapat mencetak tabel tersebut. Kita dapat menentukan ctok, tok, dll., atau menggunakan daftar (list) Table.

```
>writetable(Table,ctok=ctok,wc=5);
```

```
Variable or function Table not found.
Error in:
writetable(Table,ctok=ctok,wc=5); ...
                ^
```

Fungsi tablecol() mengembalikan nilai-nilai dari kolom-kolom tabel, melewati baris-baris dengan nilai NAN ("." dalam file), serta indeks-indeks kolom yang berisi nilai-nilai tersebut.

```
>{c,i}=tablecol(MT,[5,6]);
```

```
Variable or function MT not found.
Error in:
{c,i}=tablecol(MT,[5,6]); ...
                 ^
```

Kita dapat menggunakan ini untuk mengekstrak kolom-kolom dari tabel untuk membuat tabel baru.

```
>j=[1,5,6]; writetable(MT[i,j],labc=hd[j],ctok=[2],tok=tok)
```

```
Variable or function i not found.
Error in:
j=[1,5,6]; writetable(MT[i,j],labc=hd[j],ctok=[2],tok=tok) ...
                          ^
```

Tentu saja, dalam hal ini, kita perlu mengekstrak tabel itu sendiri dari daftar (list) Table.

```
>MT=Table[1];
```

```
Table is not a variable!
Error in:
MT=Table[1]; ...
           ^
```

Tentu saja, kita juga dapat menggunakannya untuk menentukan nilai rata-rata dari suatu kolom atau nilai statistik lainnya.

```
>mean(tablecol(MT,6))
```

```
Variable or function MT not found.
Error in:
mean(tablecol(MT,6)) ...
                ^
```

Fungsi getstatistics() mengembalikan elemen-elemen dalam bentuk vektor, beserta jumlah kemunculannya. Kita menggunakannya untuk nilai "m" dan "f" dalam kolom kedua tabel kita.

```
>{xu,count}=getstatistics(tablecol(MT,2)); xu, count,
```

```
Variable or function MT not found.
Error in:
{xu,count}=getstatistics(tablecol(MT,2)); xu, count, ...
                                     ^
```

Kita dapat mencetak hasilnya dalam bentuk tabel baru.

```
>writetable(count',labr=tok[xu])
```

```
Variable count not found!
Error in:
writetable(count',labr=tok[xu]) ...
                 ^
```

Fungsi selecttable() mengembalikan tabel baru dengan nilai-nilai dalam satu kolom yang dipilih dari vektor indeks. Pertama, kita mencari indeks dua nilai dalam tabel token kita.

Catatan tambahan:

Fungsi indexof(v, x) berarti bahwa mencari x dalam vektor v.

```
>v:=indexof(tok,["g","vg"])
```

```
Variable or function tok not found.
Error in:
v:=indexof(tok,["g","vg"]) ...
              ^
```

Sekarang kita dapat memilih baris-baris tabel yang memiliki salah satu dari nilai-nilai dalam vektor v di kolom kelima mereka.

```
>MT1:=MT[selectrows(MT,5,v)]; i:=sortedrows(MT1,5);
```

```
Variable or function MT not found.
Error in:
MT1:=MT[selectrows(MT,5,v)]; i:=sortedrows(MT1,5); ...
                     ^
```

Sekarang kita dapat mencetak tabel dengan nilai-nilai yang diekstrak dan diurutkan dalam kolom kelima.

```
>writetable(MT1[i],labc=hd,ctok=ctok,tok=tok,wc=7);
```

```
Variable or function i not found.
Error in:
writetable(MT1[i],labc=hd,ctok=ctok,tok=tok,wc=7); ...
                ^
```

Untuk statistik selanjutnya, kita ingin mengaitkan dua kolom dari tabel. Jadi, kita akan mengekstrak kolom 2 dan 4, lalu mengurutkan tabelnya.

```
>i=sortedrows(MT,[2,4]);  ...
>  writetable(tablecol(MT[i],[2,4])',ctok=[1,2],tok=tok)//Tabel diurutkan s
```

```
Variable or function MT not found.
Error in:
i=sortedrows(MT,[2,4]);    writetable(tablecol(MT[i],[2,4])',c ...
               ^
```

Dengan menggunakan fungsi getstatistics(), kita dapat menghubungkan jumlah kemunculan dalam dua kolom tabel satu sama lain.

```
>MT24=tablecol(MT,[2,4]); ...
>{xu1,xu2,count}=getstatistics(MT24[1],MT24[2]); ...
>writetable(count,labr=tok[xu1],labc=tok[xu2])
```

```
Variable or function MT not found.
Error in:
MT24=tablecol(MT,[2,4]); {xu1,xu2,count}=getstatistics(MT24[1] ...
                ^
```

Sebuah tabel dapat ditulis ke dalam sebuah file.

```
>filename="test.dat"; ...
>writetable(count,labr=tok[xu1],labc=tok[xu2],file=filename);
```

```
Variable or function count not found.
Error in:
filename="test.dat"; writetable(count,labr=tok[xu1],labc=tok[x ...
                                ^
```

Kemudian kita dapat membaca tabel dari file tersebut.

```
>{MT2,hd,tok2,hdr}=readtable(filename,>clabs,>rlabs); ...
>writetable(MT2,labr=hdr,labc=hd)
```

```
Could not open the file
test.dat
for reading!
Try "trace errors" to inspect local variables after errors.
readtable:
    if filename!=none then open(filename,"r"); endif;
```

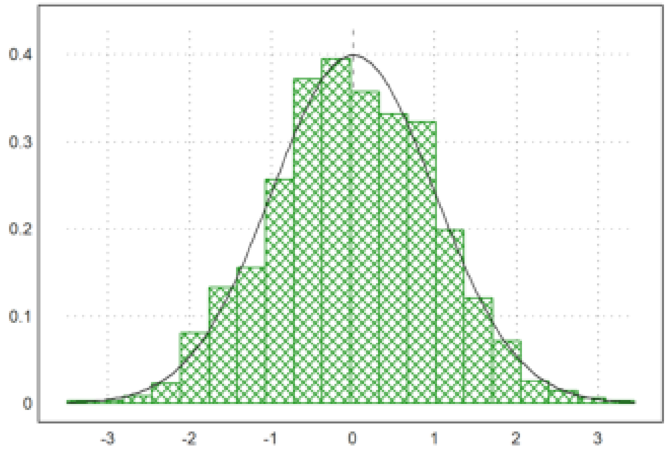
Dan menghapus file tersebut.

```
>fileremove(filename);
```

Distribusi

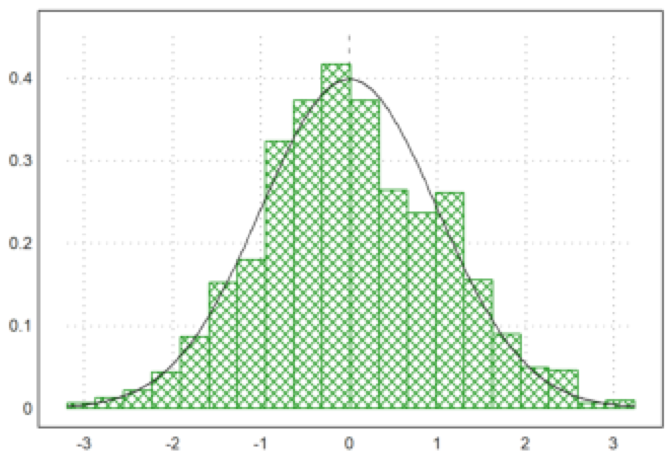
Dengan plot2d, ada metode yang sangat mudah untuk memplot distribusi data eksperimental.

```
>p=normal(1,1000); //1000 sampel acak yang terdistribusi normal p
>plot2d(p,distribution=20,style="\/"); // plot sampel acak p
>plot2d("qnormal(x,0,1)",add=1): // tambahkan plot distribusi normal standa
```



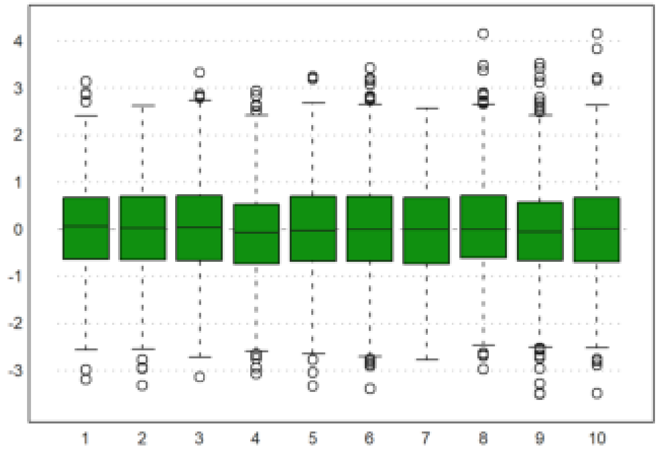
Harap perhatikan perbedaan antara diagram batang (sampel) dan kurva normal (distribusi sebenarnya). Silakan masukkan kembali tiga perintah tersebut untuk melihat hasil pengambilan sampel yang lain.

```
>p=normal(1,1000);
>plot2d(p,distribution=20,style="\/");
>plot2d("qnormal(x,0,1)",add=1):
```



Berikut adalah perbandingan dari 10 simulasi 1000 nilai yang terdistribusi secara normal menggunakan diagram kotak (box plot). Plot ini menunjukkan median, kuartil 25% dan 75%, nilai minimal dan maksimal, serta nilai-nilai yang berada di luar jangkauan (outliers).

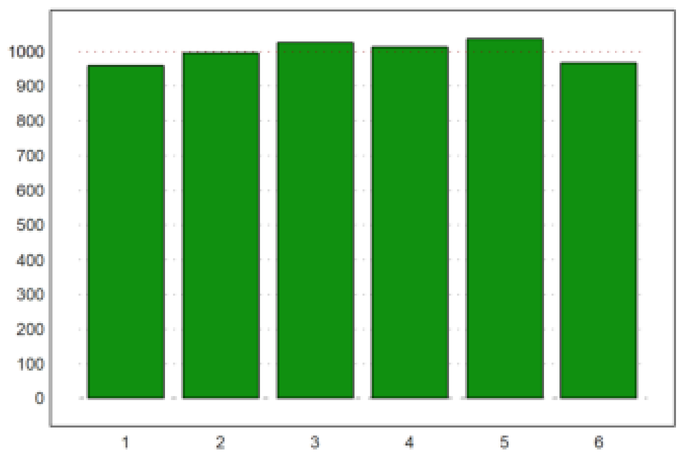
```
>p=normal(10,1000); boxplot(p):
```



Untuk menghasilkan bilangan bulat acak, Euler memiliki fungsi intrandom. Mari kita simulasi lemparan dadu dan plot distribusinya.

Kita akan menggunakan fungsi getmultiplicities(v, x) yang menghitung seberapa sering elemen-elemen dari v muncul dalam x. Kemudian, kita akan memplot hasilnya menggunakan fungsi columnsplot().

```
>k=intrandom(1,6000,6);  ...
>columnsplot(getmultiplicities(1:6,k));  ...
>ygrid(1000,color=red):
```



Catatan tambahan :
intrandom(n, m, k): Matriks dari variabel acak
getmultiplicities : Menghitung seberapa sering elemen-elemen dari x muncul dalam y.

Meskipun intrandom(n, m, k) mengembalikan bilangan bulat yang terdistribusi seragam dari 1 hingga k, kita juga dapat menggunakan distribusi bilangan bulat lainnya dengan menggunakan randpint().

Pada contoh berikut, probabilitas untuk 1, 2, 3 adalah 0,4, 0,1, 0,5 berturut-turut.

```
>randpint(1,1000,[0.4,0.1,0.5]); getmultiplicities(1:3,%)
```

```
[394,   107,   499]
```

Euler dapat menghasilkan nilai acak dari berbagai distribusi. Silakan lihat dokumentasi (reference) untuk informasi lebih lanjut.

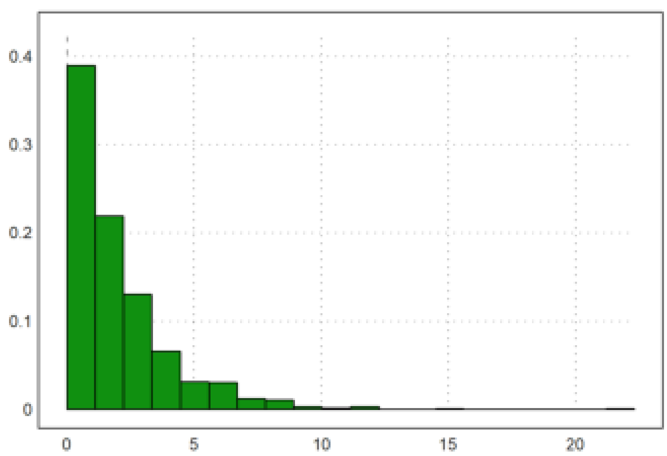
Sebagai contoh, kita akan mencoba distribusi eksponensial. Sebuah variabel acak kontinu X dikatakan memiliki distribusi eksponensial jika PDF (Probability Density Function) nya diberikan oleh:

$$f_X(x) = \lambda e^{-\lambda x}, \quad x > 0, \quad \lambda > 0,$$

dengan parameter

$$\lambda = \frac{1}{\mu}, \quad \mu \text{ is the mean, and denoted by } X \sim \text{Exponential}(\lambda).$$

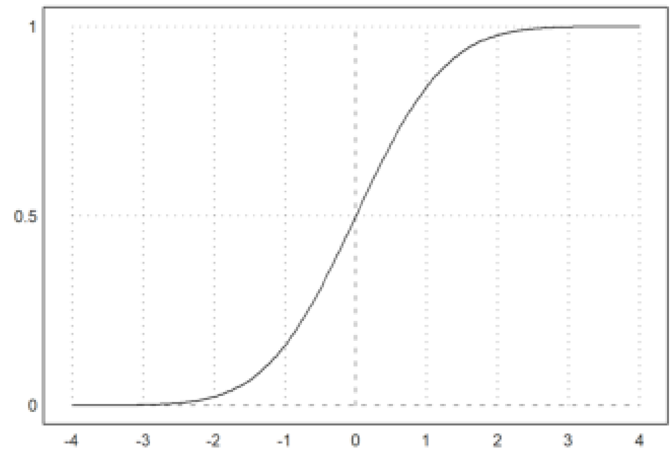
```
>plot2d(randexponential(1,1000,2),>distribution):
```



Catatan tambahan :
randexponential : matrix acak dari distribusi eksponensial

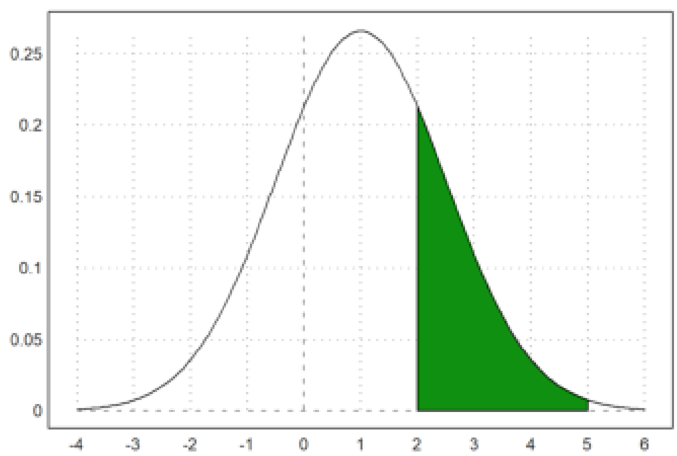
Untuk banyak distribusi, Euler dapat menghitung fungsi distribusi dan inversenya.

```
>plot2d("normaldis",-4,4):
```



Berikut adalah salah satu cara untuk memplot kuantil.

```
>plot2d("qnormal(x,1,1.5)",-4,6);  ...
>plot2d("qnormal(x,1,1.5)",a=2,b=5,>add,>filled):
```



$$\text{normaldis(x,m,d)} = \int_{-\infty}^{x} \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t-m}{d})^2}\ dt.$$

Probabilitas berada di area hijau adalah sebagai berikut.

```
>normaldis(5,1,1.5)-normaldis(2,1,1.5)
```

```
0.248662156979
```

Ini dapat dihitung secara numerik dengan integral berikut.

$$\int_2^5 \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-1}{1.5})^2} \, dx.$$

```
>gauss("qnormal(x,1,1.5)",2,5)
```

```
0.248662156979
```

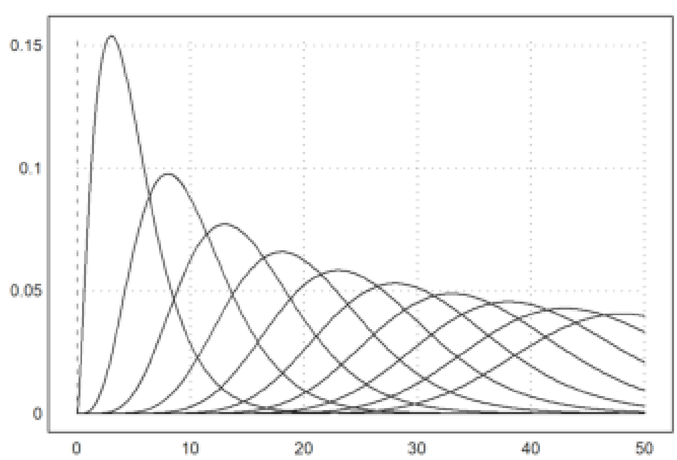
Mari bandingkan distribusi binomial dengan distribusi normal dengan rata-rata dan deviasi standar yang sama. Fungsi invbindis() menyelesaikan interpolasi linear antara nilai-nilai bulat.

```
>invbindis(0.95,1000,0.5), invnormaldis(0.95,500,0.5*sqrt(1000))
```

```
525.516721219
526.007419394
```

Fungsi qdis() adalah fungsi densitas distribusi chi-square. Seperti biasa, Euler memetakan vektor ke fungsi ini. Dengan demikian, kita dapat dengan mudah membuat plot untuk semua distribusi chi-square dengan derajat 5 hingga 30 dengan cara berikut.

```
>plot2d("qchidis(x,(5:5:50)')",0,50):
```



Euler memiliki fungsi yang akurat untuk mengevaluasi distribusi. Mari kita periksa chidis() dengan sebuah integral.

Penamaan fungsi tersebut mencoba konsisten. Misalnya,

- distribusi chi-square adalah chidis(),
- fungsi inversenya adalah invchidis(),
- fungsi densitasnya adalah qchidis().

Komplemen dari distribusi (ekor atas) disebut chicdis().

```
>chidis(1.5,2), integrate("qchidis(x,2)",0,1.5)
```

```
0.527633447259
0.527633447259
```

Distribusi Diskret

Untuk mendefinisikan distribusi diskret Anda sendiri, Anda dapat menggunakan metode berikut.

Pertama, kita atur fungsi distribusi.

```
>wd = 0|((1:6)+[-0.01,0.01,0,0,0,0])/6
```

```
[0,  0.165,  0.335,  0.5,  0.666667,  0.833333,  1]
```

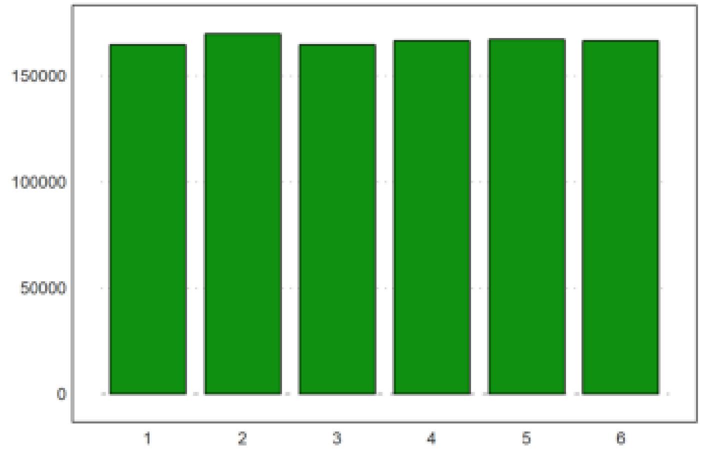
Artinya, dengan probabilitas wd[i+1] - wd[i], kita menghasilkan nilai acak i.

Ini hampir merupakan distribusi seragam. Mari kita tentukan pembangkit angka acak untuk ini. Fungsi find(v, x) mencari nilai x dalam vektor v. Fungsi ini juga berfungsi untuk vektor x.

```
>function wrongdice (n,m) := find(wd,random(n,m))
```

Kesalahan tersebut sangat halus sehingga kita hanya bisa melihatnya dengan sangat banyak iterasi.

```
>columnsplot(getmultiplicities(1:6,wrongdice(1,1000000))):
```



Berikut adalah fungsi sederhana untuk memeriksa distribusi seragam dari nilai-nilai 1 hingga K dalam vektor v. Kita menerima hasilnya jika untuk semua frekuensi

$$\left|f_i - \frac{1}{K}\right| < \frac{\delta}{\sqrt{n}}.$$

```
>function checkrandom (v, delta=1) ...
```

```
K=max(v); n=cols(v);
fr=getfrequencies(v,1:K);
return max(fr/n-1/K)<delta/sqrt(n);
endfunction
```

Memang, fungsi tersebut menolak distribusi seragam.

```
>checkrandom(wrongdice(1,1000000))
```

```
0
```

Dan itu menerima pembangkit angka acak bawaan.

```
>checkrandom(intrandom(1,1000000,6))
```

```
1
```

Kita dapat menghitung distribusi binomial. Pertama, ada fungsi binomialsum(), yang mengembalikan probabilitas i atau kurang hasil dalam n percobaan.

```
>bindis(410,1000,0.4)//Distribusi Binomial Kumulatif
```

```
0.751401349654
```

Fungsi invers Beta digunakan untuk menghitung interval kepercayaan Clopper-Pearson untuk parameter p. Tingkat kepercayaan default adalah alpha.

Makna dari interval ini adalah bahwa jika p berada di luar interval tersebut, hasil yang diamati 410 dari 1000 adalah hal yang jarang terjadi.

```
>clopperpearson(410,1000)
```

```
[0.37932,  0.441212]
```

Perintah-perintah berikut adalah cara langsung untuk mendapatkan hasil di atas. Namun, untuk nilai n yang besar, penjumlahan langsung tidak akurat dan lambat.

```
>p=0.4; i=0:410; n=1000; sum(bin(n,i)*p^i*(1-p)^(n-i))
```

```
0.751401349655
```

Sekadar informasi, invbinsum() menghitung invers dari binomialsum().

```
>invbindis(0.75,1000,0.4)
```

```
409.932733047
```

Dalam permainan Bridge, kita asumsikan ada 5 kartu istimewa (dari total 52 kartu) dalam dua tangan (26 kartu). Mari kita hitung probabilitas dari distribusi yang lebih buruk daripada 3:2 (misalnya 0:5, 1:4, 4:1, atau 5:0).

```
>2*hypergeomsum(1,5,13,26)
```

```
0.321739130435
```

Ada juga simulasi distribusi multinomial.

```
>randmultinomial(10,1000,[0.4,0.1,0.5])
```

```
            418          92         490
            445          90         465
            403         114         483
            405          95         500
            384         107         509
            414          93         493
            419          90         491
            394         101         505
            382         103         515
            403         128         469
```

Plotting Data

Untuk memplot data, kita mencoba hasil pemilihan umum Jerman sejak tahun 1990, diukur dalam kursi.

```
>BW := [ ...
>1990,662,319,239,79,8,17; ...
>1994,672,294,252,47,49,30; ...
>1998,669,245,298,43,47,36; ...
>2002,603,248,251,47,55,2; ...
>2005,614,226,222,61,51,54; ...
>2009,622,239,146,93,68,76; ...
>2013,631,311,193,0,63,64];
```

Untuk partai-partai politik, kita menggunakan string nama-nama partai.

```
>P:=["CDU/CSU","SPD","FDP","Gr","Li"];
```

Mari mencetak persentasenya dengan rapi.

Pertama, kita ekstrak kolom-kolom yang diperlukan. Kolom 3 hingga 7 adalah kursi-kursi masing-masing partai, dan kolom 2 adalah total jumlah kursi. Kolom adalah tahun pemilihan umum.

```
>BT:=BW[,3:7]; BT:=BT/sum(BT); YT:=BW[,1]';
```

Kemudian kita cetak statistiknya dalam bentuk tabel. Kita gunakan nama-nama partai sebagai judul kolom, dan tahun-tahun sebagai judul baris. Lebar default untuk kolom-kolom adalah wc=10, namun kita lebih memilih output yang lebih padat. Kolom-kolom akan diperluas untuk label-label kolom, jika diperlukan.

```
>writetable(BT*100,wc=6,dc=0,>fixed,labc=P,labr=YT)
```

```
          CDU/CSU   SPD   FDP    Gr    Li
    1990       48    36    12     1     3
    1994       44    38     7     7     4
    1998       37    45     6     7     5
    2002       41    42     8     9     0
    2005       37    36    10     8     9
    2009       38    23    15    11    12
    2013       49    31     0    10    10
```

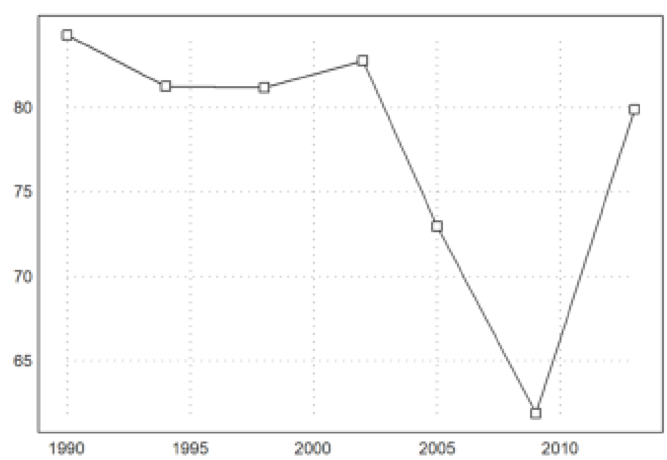
Perkalian matriks berikut mengekstrak jumlah persentase dari dua partai besar, menunjukkan bahwa partai-partai kecil telah mendapatkan perolehan kursi di parlemen hingga tahun 2009.

```
>BT1:=(BT.[1;1;0;0;0])'*100
```

```
[84.29,  81.25,  81.1659,  82.7529,  72.9642,  61.8971,  79.8732]
```

Ada juga plot statistik sederhana. Kita menggunakannya untuk menampilkan garis dan titik secara bersamaan. Alternatifnya adalah dengan memanggil plot2d dua kali dengan >add.

```
>statplot(YT,BT1,"b"):
```

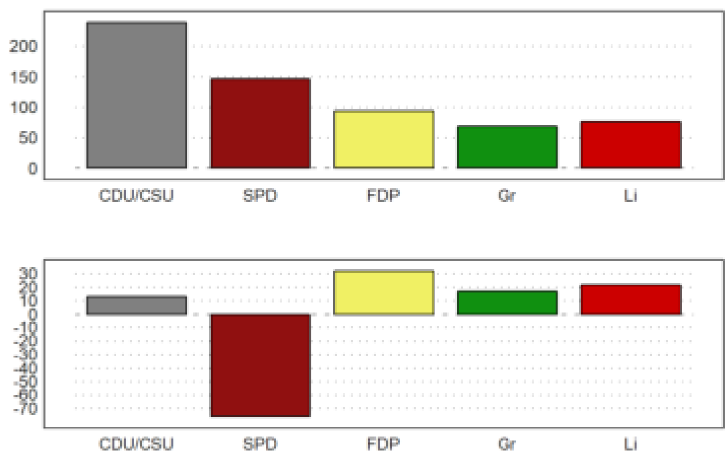


Tentukan beberapa warna untuk setiap partai.

```
>CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.8,0,0)];
```

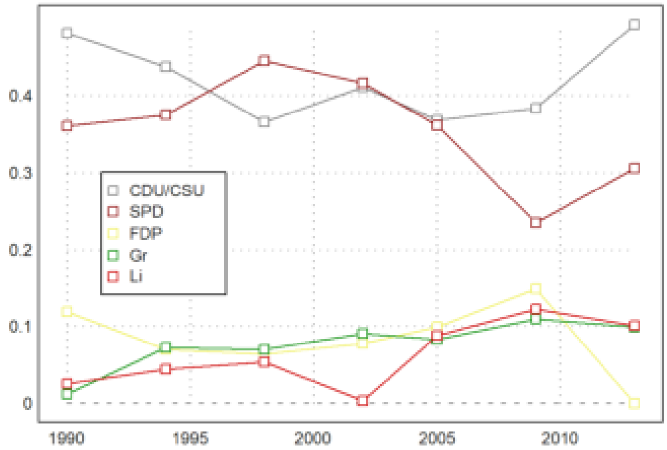
Sekarang kita dapat memplot hasil pemilihan tahun 2009 dan perubahan hasilnya dalam satu plot menggunakan fungsi figure. Kita dapat menambahkan vektor kolom ke setiap plot.

```
>figure(2,1);  ...
>figure(1); columnsplot(BW[6,3:7],P,color=CP); ...
>figure(2); columnsplot(BW[6,3:7]-BW[5,3:7],P,color=CP);  ...
>figure(0):
```



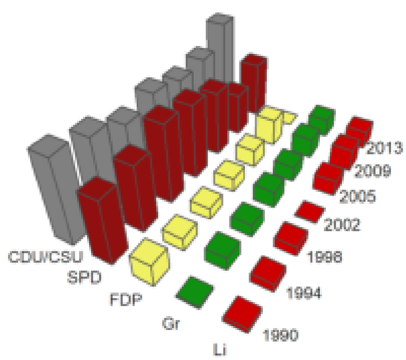
Plot data menggabungkan baris-baris data statistik dalam satu plot.

```
>J:=BW[,1]'; DP:=BW[,3:7]'; ...
>dataplot(YT,BT',color=CP);  ...
>labelbox(P,colors=CP,styles="[]",>points,w=0.2,x=0.3,y=0.4):
```



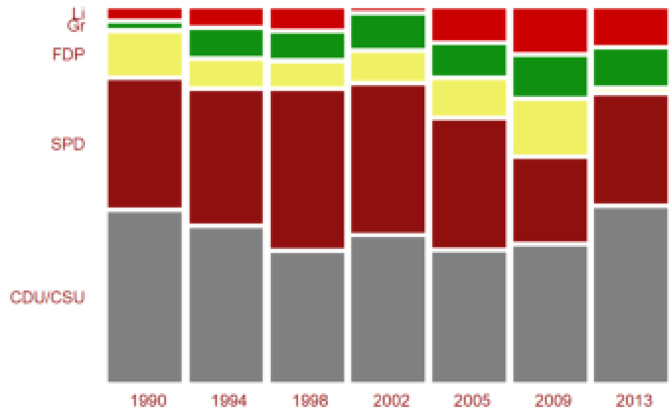
Plot kolom 3D menunjukkan baris-baris data statistik dalam bentuk kolom. Kita menyediakan label untuk baris dan kolom. Sudut (angle) adalah sudut pandang tampilan.

```
>columnsplot3d(BT,scols=P,srows=YT, ...
>  angle=30°,ccols=CP):
```



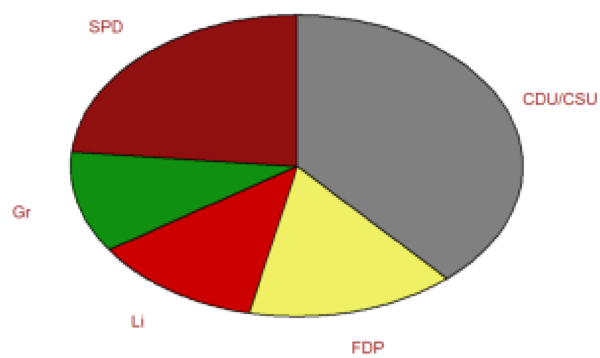
Representasi lainnya adalah plot mozaik. Perlu diingat bahwa kolom-kolom plot ini merepresentasikan kolom-kolom dari matriks di sini. Karena panjang label CDU/CSU, kita menggunakan jendela yang lebih kecil dari biasanya.

```
>shrinkwindow(>smaller);  ...
>mosaicplot(BT',srows=YT,scols=P,color=CP,style="#"); ...
>shrinkwindow():
```



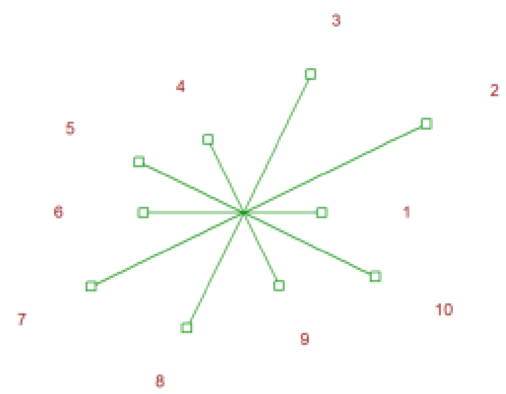
Kita juga dapat membuat diagram lingkaran (pie chart). Karena hitam dan kuning membentuk koalisi, kita akan menyusun ulang elemen-elemen tersebut.

```
>i=[1,3,5,4,2]; piechart(BW[6,3:7][i],color=CP[i],lab=P[i]):
```



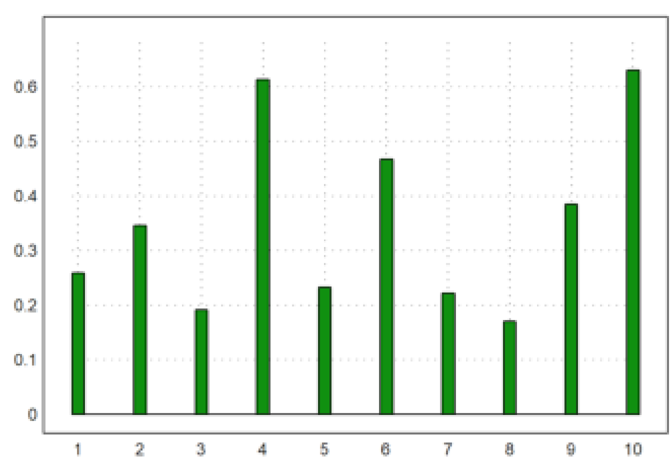
Di sini adalah jenis plot yang lain.

```
>starplot(normal(1,10)+4,lab=1:10,>rays):
```



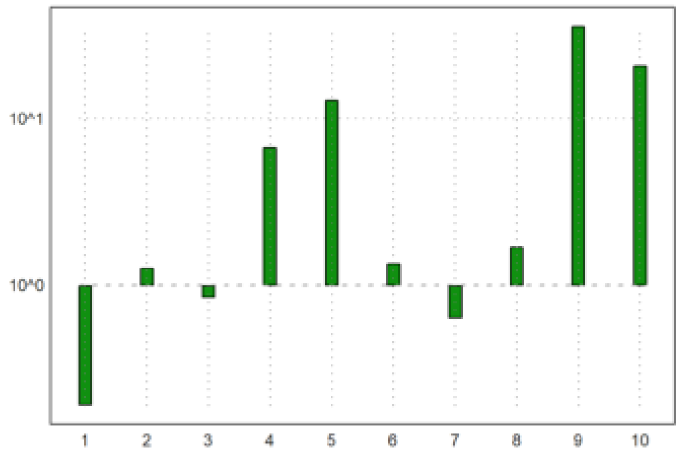
Beberapa plot dalam plot2d cocok untuk data statis. Berikut adalah plot impuls dari data acak yang terdistribusi secara merata dalam [0,1].

```
>plot2d(makeimpulse(1:10,random(1,10)),>bar):
```



But for exponentially distributed data, we may need a logarithmic plot.

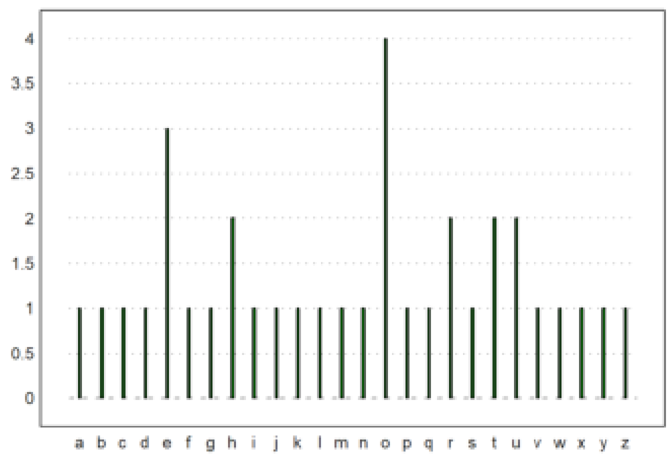
```
>logimpulseplot(1:10,-log(random(1,10))*10):
```



Fungsi ‘columnsplot()‘ lebih mudah digunakan, karena hanya memerlukan vektor nilai. Selain itu, kita dapat mengatur label sesuai keinginan kita, seperti yang telah kami tunjukkan dalam tutorial ini sebelumnya.

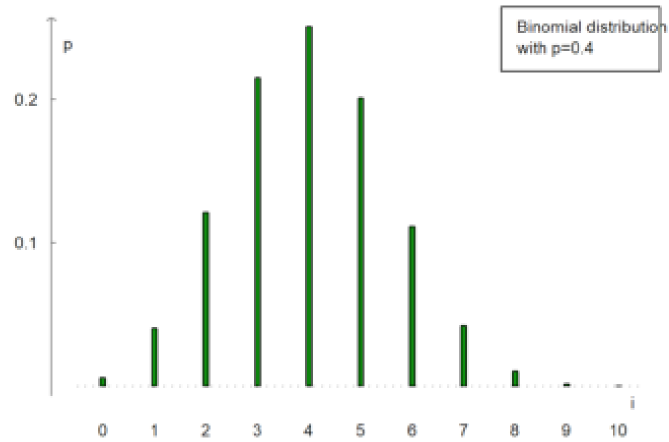
Berikut adalah aplikasi lain, di mana kita menghitung karakter dalam sebuah kalimat dan membuat plot statistiknya.

```
>v=strtochar("the quick brown fox jumps over the lazy dog"); ...
>w=ascii("a"):ascii("z"); x=getmultiplicities(w,v); ...
>cw=[]; for k=w; cw=cw|char(k); end; ...
>columnsplot(x,lab=cw,width=0.05):
```



Juga mungkin untuk secara manual mengatur sumbu-sumbu.

```
>n=10; p=0.4; i=0:n; x=bin(n,i)*p^i*(1-p)^(n-i); ...
>columnsplot(x,lab=i,width=0.05,<frame,<grid); ...
>yaxis(0,0:0.1:1,style="->",>left); xaxis(0,style="."); ...
>label("p",0,0.25), label("i",11,0); ...
>textbox(["Binomial distribution","with p=0.4"]):
```



Berikut adalah cara untuk membuat plot frekuensi angka dalam sebuah vektor.

Kita membuat vektor dari angka-angka acak integer dari 1 hingga 6.

```
>v:=intrandom(1,10,10)
```

```
[4,  5,  2,  6,  1,  10,  8,  4,  1,  2]
```

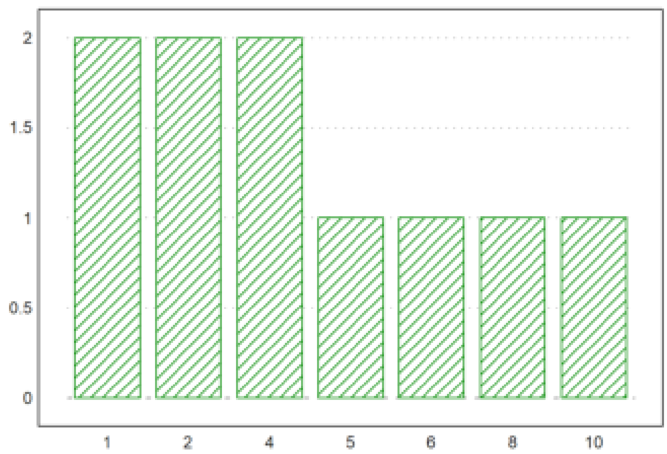
Kemudian ekstrak angka-angka unik dalam vektor v.

```
>vu:=unique(v)
```

```
[1,  2,  4,  5,  6,  8,  10]
```

Dan plot frekuensi tersebut dalam sebuah plot kolom.

```
>columnsplot(getmultiplicities(vu,v),lab=vu,style="/"):
```



Kami ingin mendemonstrasikan fungsi-fungsi untuk distribusi empiris dari nilai-nilai.

```
>x=normal(1,20);
```

Fungsi ‘empdist(x, vs)‘ memerlukan sebuah larik nilai yang telah diurutkan. Oleh karena itu, kita perlu mengurutkan x sebelum kita dapat menggunakannya.

```
>xs=sort(x);
```

Kemudian kita membuat plot distribusi empiris dan beberapa batang kepadatan dalam satu plot. Kali ini, daripada menggunakan plot batang untuk distribusi, kita akan menggunakan plot gigi gergaji (sawtooth plot).

```
>figure(2,1); ...
>figure(1); plot2d("empdist",-4,4;xs); ...
>figure(2); plot2d(histo(x,v=-4:0.2:4,<bar));  ...
>figure(0):
```

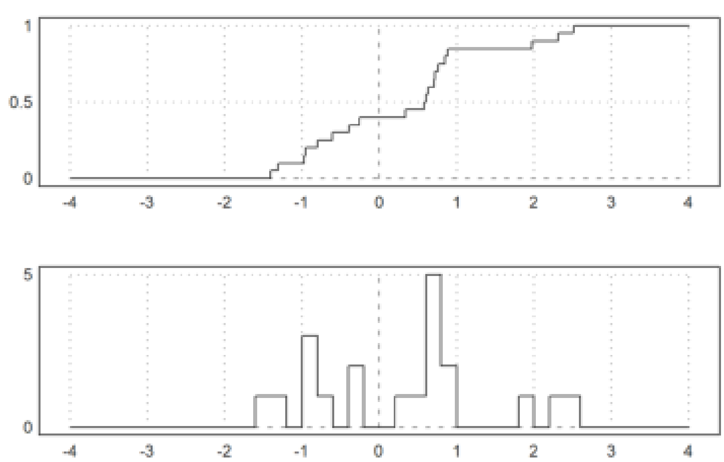
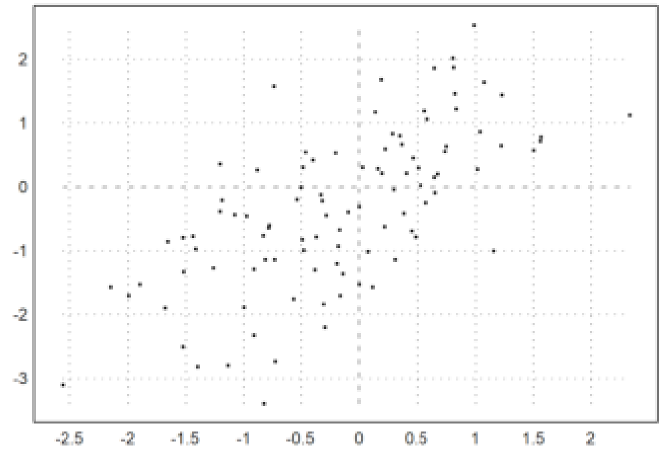


Diagram hamburan (scatter plot) mudah dilakukan dalam Euler dengan plot titik biasa. Grafik berikut menunjukkan bahwa X dan X+Y jelas berkorelasi positif.

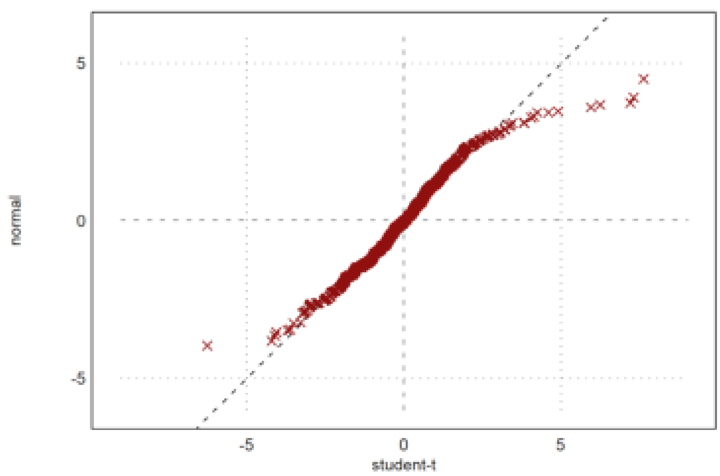
```
>x=normal(1,100); plot2d(x,x+rotright(x),>points,style=".."):
```



Seringkali, kita ingin membandingkan dua sampel dengan distribusi yang berbeda. Hal ini dapat dilakukan dengan menggunakan plot kuantil-kuantil (quantile-quantile plot).

Untuk sebuah uji, kita mencoba distribusi t-student dan distribusi eksponensial.

```
>x=randt(1,1000,5); y=randnormal(1,1000,mean(x),dev(x)); ...
>plot2d("x",r=6,style="--",yl="normal",xl="student-t",>vertical); ...
>plot2d(sort(x),sort(y),>points,color=red,style="x",>add):
```



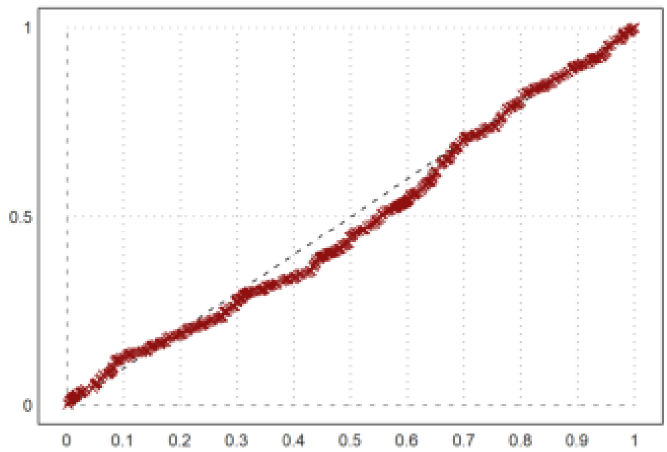
Grafik tersebut dengan jelas menunjukkan bahwa nilai yang terdistribusi secara normal cenderung lebih kecil di ujung-ujung ekstrem.

Jika kita memiliki dua distribusi dengan ukuran yang berbeda, kita dapat memperbesar yang lebih kecil atau mengecilkan yang lebih besar. Fungsi berikut baik digunakan untuk keduanya. Ini mengambil nilai median dengan persentase antara 0 dan 1.

```
>function medianexpand (x,n) := median(x,p=linspace(0,1,n-1));
```

Mari kita bandingkan dua distribusi yang sama.

```
>x=random(1000); y=random(400); ...
>plot2d("x",0,1,style="--"); ...
>plot2d(sort(medianexpand(x,400)),sort(y),>points,color=red,style="x",>add)
```



Regresi dan Korelasi

Regresi linear dapat dilakukan dengan menggunakan fungsi polyfit() atau berbagai fungsi fit lainnya.

Untuk memulai, kita dapat menemukan garis regresi untuk data univariat dengan polyfit(x, y, 1).

```
>x=1:10; y=[2,3,1,5,6,3,7,8,9,8]; writetable(x'|y',labc=["x","y"])
```

```
        x         y
        1         2
        2         3
        3         1
```

```
  4        5
  5        6
  6        3
  7        7
  8        8
  9        9
 10        8
```

Kita ingin membandingkan hasil regresi tanpa bobot (non-weighted) dan dengan bobot (weighted). Pertama-tama, mari lihat koefisien regresi linearnya.

```
>p=polyfit(x,y,1)
```

```
[0.733333,  0.812121]
```

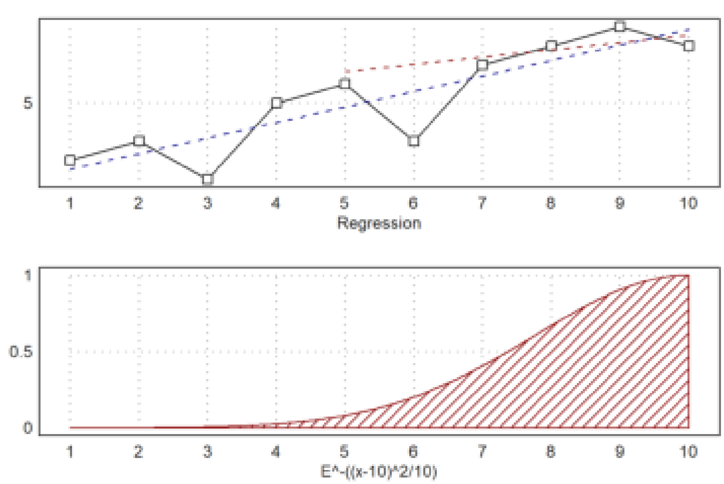
Sekarang mari lihat koefisien dengan bobot yang menekankan nilai-nilai terakhir.

```
>w &= "exp(-(x-10)^2/10)"; pw=polyfit(x,y,1,w=w(x))
```

```
[4.71566,  0.38319]
```

Kita gabungkan semuanya dalam satu plot untuk titik-titik data, garis regresi, dan untuk bobot yang digunakan.

```
>figure(2,1);  ...
>figure(1); statplot(x,y,"b",xl="Regression"); ...
>  plot2d("evalpoly(x,p)",>add,color=blue,style="--"); ...
>  plot2d("evalpoly(x,pw)",5,10,>add,color=red,style="--"); ...
>figure(2); plot2d(w,1,10,>filled,style="/",fillcolor=red,xl=w); ...
>figure(0):
```



Untuk contoh lain, kita membaca hasil survei tentang mahasiswa, usia mereka, usia orang tua mereka, dan jumlah saudara kandung dari sebuah file.

Tabel ini berisi "m" dan "f" dalam kolom kedua. Kita menggunakan variabel tok2 untuk mengatur terjemahan yang sesuai daripada membiarkan readtable() mengumpulkan terjemahan.

```
>{MS,hd}:=readtable("table1.dat",tok2:=["m","f"]);  ...
>writetable(MS,labc=hd,tok2:=["m","f"]);
```

```
Could not open the file
table1.dat
for reading!
Try "trace errors" to inspect local variables after errors.
readtable:
    if filename!=none then open(filename,"r"); endif;
```

Bagaimana usia-usia ini bergantung satu sama lain? Kesimpulan awal dapat diperoleh dari scatterplot pasangan (pairwise scatterplot).

```
>scatterplots(tablecol(MS,3:5),hd[3:5]):
```

```
Variable or function MS not found.
Error in:
scatterplots(tablecol(MS,3:5),hd[3:5]): ...
                          ^
```

Jelas bahwa usia ayah dan ibu saling bergantung. Mari kita tentukan dan plot garis regresinya.

```
>cs:=MS[,4:5]'; ps:=polyfit(cs[1],cs[2],1)
```

```
MS is not a variable!
Error in:
cs:=MS[,4:5]'; ps:=polyfit(cs[1],cs[2],1) ...
            ^
```

Ini jelas bukan model yang tepat. Garis regresinya seharusnya adalah s = 17 + 0.74t, di mana t adalah usia ibu dan s adalah usia ayah. Perbedaan usia mungkin sedikit bergantung pada usia, tetapi tidak sebanyak itu.

Sebaliknya, kami mencurigai bahwa fungsi seperti s = a + t. Kemudian, a adalah rata-rata dari s-t. Itu adalah perbedaan usia rata-rata antara ayah dan ibu.

```
>da:=mean(cs[2]-cs[1])
```

```
cs is not a variable!
Error in:
da:=mean(cs[2]-cs[1]) ...
              ^
```

Mari kita plot ini dalam satu scatter plot.

```
>plot2d(cs[1],cs[2],>points);  ...
>plot2d("evalpoly(x,ps)",color=red,style=".",>add);  ...
>plot2d("x+da",color=blue,>add):
```

```
cs is not a variable!
Error in:
plot2d(cs[1],cs[2],>points);  plot2d("evalpoly(x,ps)",color=re ...
            ^
```

Berikut adalah box plot dari dua usia. Ini hanya menunjukkan bahwa usia-usia tersebut berbeda.

```
>boxplot(cs,["mothers","fathers"]):
```

```
Variable or function cs not found.
Error in:
boxplot(cs,["mothers","fathers"]): ...
          ^
```

Menarik bahwa perbedaan median tidak sebesar perbedaan mean.

```
>median(cs[2])-median(cs[1])
```

```
cs is not a variable!
Error in:
median(cs[2])-median(cs[1]) ...
            ^
```

Koefisien korelasi menunjukkan adanya korelasi positif.

```
>correl(cs[1],cs[2])
```

```
cs is not a variable!
Error in:
correl(cs[1],cs[2]) ...
            ^
```

Korelasi peringkat adalah ukuran untuk urutan yang sama dalam kedua vektor. Juga sangat positif.

```
>rankcorrel(cs[1],cs[2])
```

```
cs is not a variable!
Error in:
rankcorrel(cs[1],cs[2]) ...
                 ^
```

Membuat Fungsi Baru

Tentu saja, bahasa EMT dapat digunakan untuk membuat fungsi-fungsi baru. Misalnya, kita dapat mendefinisikan fungsi kesarjanaan (skewness).

$$\text{sk}(x) = \frac{\sqrt{n} \sum_i (x_i - m)^3}{\left(\sum_i (x_i - m)^2\right)^{3/2}}$$

di mana m adalah rata-rata dari x.

```
>function skew (x:vector) ...
```

```
m=mean(x);
return sqrt(cols(x))*sum((x-m)^3)/(sum((x-m)^2))^(3/2);
endfunction
```

Seperti yang Anda lihat, kita dengan mudah dapat menggunakan bahasa matriks untuk mendapatkan implementasi yang sangat singkat dan efisien. Mari kita coba fungsi ini.

```
>data=normal(20); skew(normal(10))
```

```
-0.0794571577159
```

Berikut adalah fungsi lain yang disebut koefisien skewness Pearson.

```
>function skew1 (x) := 3*(mean(x)-median(x))/dev(x)
>skew1(data)
```

```
-1.04418407161
```

Simulasi Monte Carlo

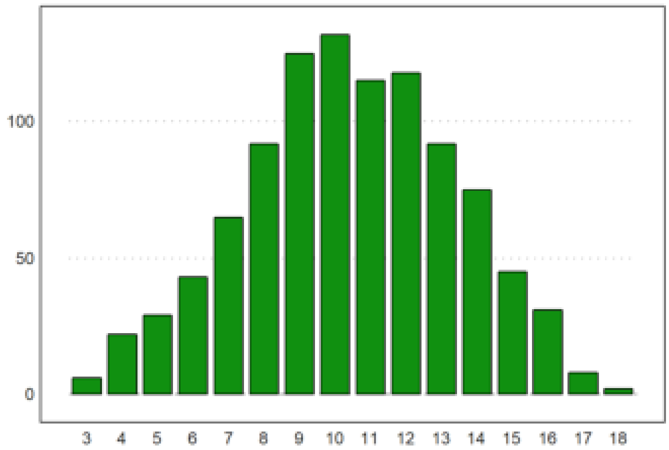
Euler dapat digunakan untuk mensimulasikan peristiwa acak. Kami telah melihat contoh-contoh sederhana di atas. Berikut adalah contoh lain yang mensimulasikan 1000 kali lemparan tiga dadu, dan menghitung distribusi jumlahnya.

```
>ds:=sum(intrandom(1000,3,6))'; fs=getmultiplicities(3:18,ds)
```

```
[6,  22,  29,  43,  65,  92,  125,  132,  115,  118,  92,  75,  45,
31,  8,  2]
```

Sekarang kita dapat membuat plot hasil simulasi ini.

```
>columnsplot(fs,lab=3:18):
```



Untuk menentukan distribusi yang diharapkan tidaklah mudah. Kita menggunakan rekursi tingkat lanjut untuk ini.

Fungsi berikut menghitung jumlah cara di mana angka k dapat direpresentasikan sebagai jumlah dari n angka dalam rentang 1 hingga m. Ini bekerja secara rekursif dengan cara yang jelas.

```
>function map countways (k; n, m) ...
```

```
    if n==1 then return k>=1 && k<=m
    else
      sum=0;
      loop 1 to m; sum=sum+countways(k-#,n-1,m); end;
      return sum;
    end;
  endfunction
```

Berikut adalah hasilnya untuk tiga lemparan dadu.

```
>countways(5:25,5,5)
```

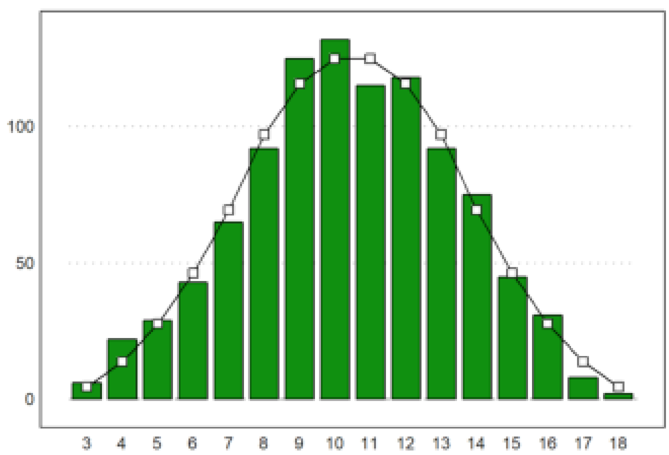
```
[1,  5,  15,  35,  70,  121,  185,  255,  320,  365,  381,  365,  320,
255,  185,  121,  70,  35,  15,  5,  1]
```

```
>cw=countways(3:18,3,6)
```

```
[1,  3,  6,  10,  15,  21,  25,  27,  27,  25,  21,  15,  10,  6,  3,
1]
```

Kita tambahkan nilai-nilai yang diharapkan ke dalam plot.

```
>plot2d(cw/6^3*1000,>add); plot2d(cw/6^3*1000,>points,>add):
```



Untuk simulasi lainnya, deviasi nilai rata-rata dari n variabel acak yang terdistribusi normal antara 0 dan 1 adalah 1/sqrt(n).

```
>longformat; 1/sqrt(10)
```

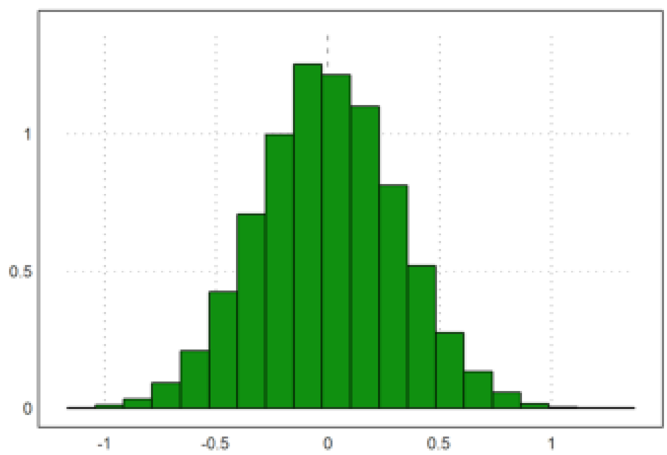
```
0.316227766017
```

Mari kita periksa ini dengan simulasi. Kami menghasilkan 10.000 kali 10 vektor acak.

```
>M=normal(10000,10); dev(mean(M)')
```

```
0.314944511075
```

```
>plot2d(mean(M)',>distribution):
```



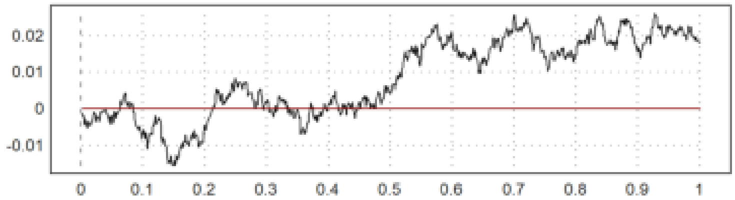
Median dari 10 angka acak yang terdistribusi normal antara 0 dan 1 memiliki deviasi yang lebih besar.

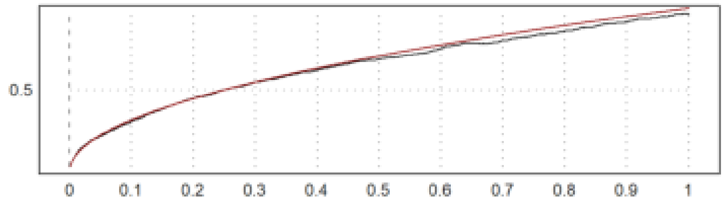
```
>dev(median(M)')
```

```
0.370541708221
```

Karena kita dapat dengan mudah menghasilkan perjalanan acak (random walks), kita dapat mensimulasikan proses Wiener. Kita ambil 1000 langkah dari 1000 proses ini. Kemudian kita membuat plot dari simpangan baku (standard deviation) dan rata-rata langkah ke-n dari proses-proses ini bersama dengan nilai-nilai yang diharapkan dalam warna merah.

```
>n=1000; m=1000; M=cumsum(normal(n,m)/sqrt(m)); ...
>t=(1:n)/n; figure(2,1); ...
>figure(1); plot2d(t,mean(M')'); plot2d(t,0,color=red,>add); ...
>figure(2); plot2d(t,dev(M')'); plot2d(t,sqrt(t),color=red,>add); ...
>figure(0):
```





Uji Statistik

Uji statistik merupakan alat penting dalam statistik. Dalam Euler, banyak uji statistik telah diimplementasikan. Semua uji ini mengembalikan kesalahan yang kita terima jika kita menolak hipotesis nol.

Sebagai contoh, kita menguji lemparan dadu untuk distribusi seragam. Pada 600 lemparan, kita mendapatkan nilai-nilai berikut, yang kita masukkan ke dalam uji chi-kuadrat.

```
>chitest([90,103,114,101,103,89],dup(100,6)')
```

```
0.498830517952
```

Uji chi-kuadrat juga memiliki mode yang menggunakan simulasi Monte Carlo untuk menguji statistik. Hasilnya seharusnya hampir sama. Parameter >p menginterpretasikan vektor y sebagai vektor probabilitas.

```
>chitest([90,103,114,101,103,89],dup(1/6,6)',>p,>montecarlo)
```

```
0.507
```

Kesalahan ini jauh terlalu besar. Jadi kita tidak bisa menolak distribusi seragam. Ini tidak membuktikan bahwa dadu kita adil. Tapi kita tidak bisa menolak hipotesis kita.

Selanjutnya, kita menghasilkan 1000 lemparan dadu menggunakan pembangkit bilangan acak, dan melakukan uji yang sama.

```
>n=1000; t=random([1,n*6]); chitest(count(t*6,6),dup(n,6)')
```

```
0.247506458901
```

Mari kita uji untuk nilai rata-rata 100 dengan uji t.

```
>s=200+normal([1,100])*10; ...
>ttest(mean(s),dev(s),100,200)
```

```
0.261148883454
```

Fungsi ttest() memerlukan nilai rata-rata, deviasi, jumlah data, dan nilai rata-rata yang akan diuji.

Sekarang mari kita periksa dua pengukuran untuk nilai rata-rata yang sama. Kita menolak hipotesis bahwa mereka memiliki nilai rata-rata yang sama jika hasilnya <0,05.

```
>tcomparedata(normal(1,10),normal(1,10))
```

```
0.116475902968
```

Jika kita menambahkan bias pada salah satu distribusi, kita akan mendapatkan lebih banyak penolakan. Ulangi simulasi ini beberapa kali untuk melihat efeknya.

```
>tcomparedata(normal(1,10),normal(1,10)+2)
```

```
5.770990267e-05
```

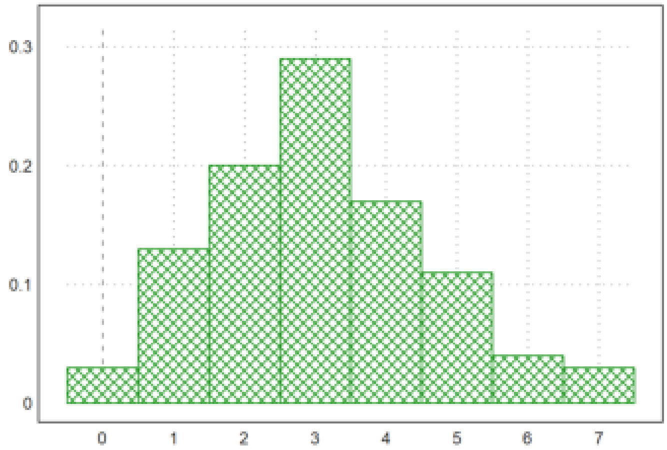
Pada contoh berikut, kita menghasilkan 20 lemparan dadu acak 100 kali dan menghitung jumlah angka satu (1) dalamnya. Rata-rata seharusnya adalah 20/6 = 3,3 angka satu.

```
>R=random(100,20); R=sum(R*6<=1)'; mean(R)
```

```
3.08
```

Sekarang kita membandingkan jumlah angka satu dengan distribusi binomial. Pertama-tama, kita membuat plot distribusi angka satu.

```
>plot2d(R,distribution=max(R)+1,even=1,style="\/"):
```




```
>t=count(R,21);
```

Kemudian kita menghitung nilai-nilai yang diharapkan.

```
>n=0:20; b=bin(20,n)*(1/6)^n*(5/6)^(20-n)*100;
```

Kita harus mengumpulkan beberapa angka untuk mendapatkan kategori-kategori yang cukup besar.

```
>t1=sum(t[1:2])|t[3:7]|sum(t[8:21]); ...
>b1=sum(b[1:2])|b[3:7]|sum(b[8:21]);
```

Uji chi-kuadrat menolak hipotesis bahwa distribusi kita adalah distribusi binomial, jika hasilnya <0,05.

```
>chitest(t1,b1)
```

```
0.717403213286
```

Contoh berikut berisi hasil dari dua kelompok orang (pria dan wanita, misalnya) yang memilih salah satu dari enam partai.

```
>A=[23,37,43,52,64,74;27,39,41,49,63,76];  ...
>  writetable(A,wc=6,labr=["m","f"],labc=1:6)
```

```
           1     2     3     4     5     6
      m   23    37    43    52    64    74
      f   27    39    41    49    63    76
```

Kita ingin menguji independensi suara dari jenis kelamin. Uji tabel chi-kuadrat melakukannya. Hasilnya terlalu besar untuk menolak independensi. Jadi, dari data ini, kita tidak bisa mengatakan apakah pemilihan bergantung pada jenis kelamin atau tidak.

```
>tabletest(A)
```

```
0.990701632326
```

Berikut adalah tabel yang diharapkan jika kita mengasumsikan frekuensi yang diamati dalam pemilihan.

```
>writetable(expectedtable(A),wc=6,dc=1,labr=["m","f"],labc=1:6)
```

```
          1     2     3     4     5     6
    m  24.9  37.9  41.9  50.3  63.3  74.7
    f  25.1  38.1  42.1  50.7  63.7  75.3
```

Kita dapat menghitung koefisien kontingensi yang dikoreksi. Karena nilai yang sangat mendekati 0, kita dapat menyimpulkan bahwa pemilihan tidak bergantung pada jenis kelamin.

```
>contingency(A)
```

```
0.0427225484717
```

Beberapa Uji Lainnya

Selanjutnya, kita menggunakan analisis varians (uji F) untuk menguji tiga sampel data yang terdistribusi normal untuk nilai rata-rata yang sama. Metode ini disebut ANOVA (analisis varians). Dalam Euler, digunakan fungsi varanalysis().

```
>x1=[109,111,98,119,91,118,109,99,115,109,94]; mean(x1),
```

```
106.545454545
```

```
>x2=[120,124,115,139,114,110,113,120,117]; mean(x2),
```

```
119.111111111
```

```
>x3=[120,112,115,110,105,134,105,130,121,111]; mean(x3)
```

```
116.3
```

```
>varanalysis(x1,x2,x3)
```

```
0.0138048221371
```

Artinya, kita menolak hipotesis nilai rata-rata yang sama. Kita melakukannya dengan probabilitas kesalahan sebesar 1,3%.

Ada juga uji median, yang menolak sampel data dengan distribusi rata-rata yang berbeda dengan menguji median dari sampel yang digabungkan.

```
>a=[56,66,68,49,61,53,45,58,54];
>b=[72,81,51,73,69,78,59,67,65,71,68,71];
>mediantest(a,b)
```

```
0.0241724220052
```

Uji lain untuk kesetaraan adalah uji peringkat. Ini jauh lebih tajam daripada uji median.

```
>ranktest(a,b)
```

```
0.00199969612469
```

Pada contoh berikut, kedua distribusi memiliki nilai rata-rata yang sama.

```
>ranktest(random(1,100),random(1,50)*3-1)
```

```
0.0908472946126
```

Mari kita coba mensimulasikan dua perawatan a dan b yang diberikan kepada orang-orang yang berbeda.

```
>a=[8.0,7.4,5.9,9.4,8.6,8.2,7.6,8.1,6.2,8.9];
>b=[6.8,7.1,6.8,8.3,7.9,7.2,7.4,6.8,6.8,8.1];
```

Uji signum (signum test) menentukan apakah a lebih baik daripada b.

```
>signtest(a,b)
```

```
0.0546875
```

Ini terlalu tinggi tingkat kesalahan. Kita tidak bisa menolak bahwa a sama baiknya dengan b.

Uji Wilcoxon lebih tajam daripada uji ini, tetapi bergantung pada nilai kuantitatif dari perbedaan.

```
>wilcoxon(a,b)
```

```
0.0296680599405
```

Mari kita mencoba dua uji lainnya menggunakan rangkaian data yang dihasilkan.

```
>wilcoxon(normal(1,20),normal(1,20)-1)
```

```
0.00499819423799
```

```
>wilcoxon(normal(1,20),normal(1,20))
```

```
0.559353645673
```

Bilangan Acak

Berikut adalah uji untuk generator bilangan acak. Euler menggunakan generator yang sangat baik, jadi kita tidak perlu mengharapkan masalah.

Pertama-tama kita menghasilkan sepuluh juta bilangan acak dalam rentang [0,1].

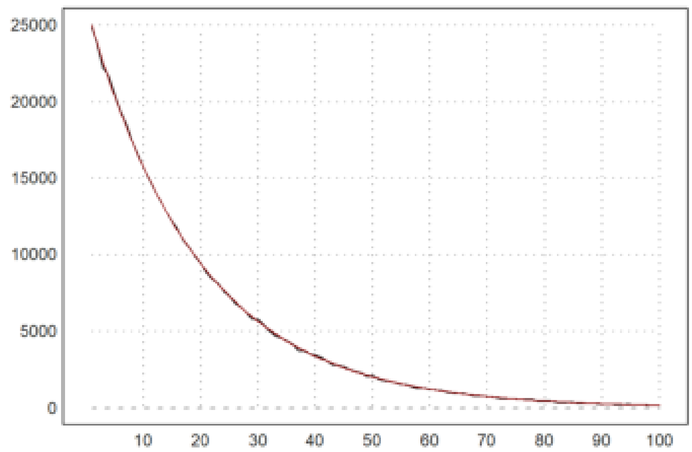
```
>n:=10000000; r:=random(1,n);
```

Selanjutnya, kita menghitung jarak antara dua bilangan yang kurang dari 0,05.

```
>a:=0.05; d:=differences(nonzeros(r<a));
```

Akhirnya, kita membuat plot jumlah kali setiap jarak terjadi, dan membandingkannya dengan nilai yang diharapkan.

```
>m=getmultiplicities(1:100,d); plot2d(m); ...
>  plot2d("n*(1-a)^(x-1)*a^2",color=red,>add):
```



Hapus data.

```
>remvalue n;
```

Pengantar untuk Pengguna Proyek R

Jelas, EMT tidak bersaing dengan R sebagai paket statistik. Namun, ada banyak prosedur statistik dan fungsi yang tersedia dalam EMT juga. Jadi, EMT dapat memenuhi kebutuhan dasar. Pada akhirnya, EMT dilengkapi dengan paket-paket numerik dan sistem aljabar komputer.

Buku catatan ini untuk Anda jika Anda akrab dengan R, tetapi perlu mengetahui perbedaan dalam sintaks antara EMT dan R. Kami akan memberikan gambaran tentang hal-hal yang jelas dan kurang jelas yang perlu Anda ketahui.

Selain itu, kami akan mengeksplorasi cara pertukaran data antara kedua sistem ini.

Harap dicatat bahwa ini adalah sebuah proyek yang masih dalam proses pengembangan.

Sintaks Dasar

Hal pertama yang Anda pelajari dalam R adalah membuat vektor. Di EMT, perbedaan utamanya adalah operator : dapat mengambil langkah. Selain itu, operator : memiliki prioritas yang rendah.

```
>n=10; 0:n/20:n-1
```

```
[0,  0.5,  1,  1.5,  2,  2.5,  3,  3.5,  4,  4.5,  5,  5.5,  6,  6.5,
7,  7.5,  8,  8.5,  9]
```

Fungsi c() tidak ada. Namun, Anda dapat menggunakan vektor untuk menggabungkan hal-hal.

Contoh berikut, seperti banyak contoh lainnya, diambil dari "Pengantar ke R" yang disertakan dalam proyek R. Jika Anda membaca PDF tersebut, Anda akan menemukan bahwa saya mengikuti jalurnya dalam tutorial ini.

```
>x=[10.4, 5.6, 3.1, 6.4, 21.7]; [x,0,x]
```

```
[10.4,  5.6,  3.1,  6.4,  21.7,  0,  10.4,  5.6,  3.1,  6.4,  21.7]
```

Operator titik dua (colon operator) dengan langkah di EMT digantikan oleh fungsi seq() di R. Kita bisa menulis fungsi ini di EMT.

```
>function seq(a,b,c) := a:b:c; ...
>seq(0,-0.1,-1)
```

```
[0,  -0.1,  -0.2,  -0.3,  -0.4,  -0.5,  -0.6,  -0.7,  -0.8,  -0.9,  -1]
```

Fungsi rep() dalam R tidak ada di EMT. Untuk input berupa vektor, fungsi ini dapat ditulis sebagai berikut.

```
>function rep(x:vector,n:index) := flatten(dup(x,n)); ...
>rep(x,2)
```

```
[10.4,  5.6,  3.1,  6.4,  21.7,  10.4,  5.6,  3.1,  6.4,  21.7]
```

Perlu dicatat bahwa "=" atau ":=" digunakan untuk penugasan dalam EMT. Operator "->" digunakan untuk satuan dalam EMT.

```
>125km -> " miles"
```

```
77.6713990297 miles
```

Operator "<-" untuk penugasan memang membingungkan dan bukan ide yang baik dalam R. Berikut akan membandingkan a dengan -4 dalam EMT.

```
>a=2; a<-4
```

```
0
```

Di R, "a <- 4 < 3" berfungsi, tetapi "a <- 4 <- 3" tidak. Saya juga memiliki ambigu yang serupa di EMT, tetapi mencoba mengatasinya sedikit demi sedikit.

EMT dan R memiliki vektor tipe boolean. Namun, di EMT, angka 0 dan 1 digunakan untuk mewakili false dan true. Di R, nilai true dan false tetap dapat digunakan dalam aritmatika biasa seperti di EMT.

```
>x<5, %*x
```

```
[0,  0,  1,  0,  0]
[0,  0,  3.1,  0,  0]
```

EMT dapat menghasilkan kesalahan atau mengembalikan NAN tergantung pada pengaturan flag "errors".

```
>errors off; 0/0, isNAN(sqrt(-1)), errors on;
```

```
NAN
1
```

String (tali) adalah sama dalam R dan EMT. Keduanya berada dalam bahasa yang berlaku, bukan dalam Unicode.

Di R, ada paket untuk Unicode. Di EMT, sebuah string dapat menjadi string Unicode. Sebuah string Unicode dapat diterjemahkan ke dalam enkodean lokal dan sebaliknya. Selain itu, u"..." dapat mengandung entitas HTML.

```
>u"&#169; Ren&eacut; Grothmann"
```

```
© René Grothmann
```

Berikut mungkin atau mungkin tidak ditampilkan dengan benar di sistem Anda sebagai A dengan titik di atas dan tanda garis di atasnya. Ini tergantung pada font yang Anda gunakan.

```
>chartoutf([480])
```

Penggabungan string dilakukan dengan "+" atau "|". Ini dapat mencakup angka, yang akan mencetak dalam format yang berlaku.

```
>"pi = "+pi
```

```
pi = 3.14159265359
```

Indeks

Sebagian besar waktu, ini akan berfungsi seperti dalam R.

Namun, EMT akan menginterpretasikan indeks negatif dari belakang vektor, sementara R menginterpretasikan x[n] sebagai x tanpa elemen ke-n.

```
>x, x[1:3], x[-2]
```

```
[10.4,   5.6,   3.1,   6.4,   21.7]
[10.4,   5.6,   3.1]
6.4
```

Perilaku R dapat dicapai dalam EMT dengan menggunakan drop().

```
>drop(x,2)
```

```
[10.4,   3.1,   6.4,   21.7]
```

Vektor logika tidak diperlakukan secara berbeda sebagai indeks dalam EMT, berbeda dengan R. Anda perlu mengekstraksi elemen-elemen yang bukan nol terlebih dahulu di EMT.

```
>x, x>5, x[nonzeros(x>5)]
```

```
[10.4,   5.6,   3.1,   6.4,   21.7]
[1,   1,   0,   1,   1]
[10.4,   5.6,   6.4,   21.7]
```

Sama seperti dalam R, vektor indeks dapat mengandung pengulangan.

```
>x[[1,2,2,1]]
```

```
[10.4,   5.6,   5.6,   10.4]
```

Tetapi penggunaan nama untuk indeks tidak mungkin di EMT. Untuk paket statistik, ini mungkin seringkali diperlukan untuk memudahkan akses ke elemen-elemen vektor.

Untuk meniru perilaku ini, kita dapat mendefinisikan fungsi seperti berikut.

```
>function sel (v,i,s) := v[indexof(s,i)]; ...
>s=["first","second","third","fourth"]; sel(x,["first","third"],s)
```

```
Trying to overwrite protected function sel!
Error in:
function sel (v,i,s) := v[indexof(s,i)]; ... ...
             ^

Trying to overwrite protected function sel!
Error in:
function sel (v,i,s) := v[indexof(s,i)]; ... ...
             ^

Trying to overwrite protected function sel!
Error in:
function sel (v,i,s) := v[indexof(s,i)]; ... ...
             ^
[10.4,  3.1]
```

Tipe Data

EMT memiliki lebih banyak tipe data yang telah ditentukan daripada R. Dalam R, jelas terdapat vektor yang bisa tumbuh. Anda dapat mengatur vektor numerik kosong v dan memberikan nilai ke elemen v[17]. Hal ini tidak mungkin dilakukan di EMT.

Berikut ini sedikit tidak efisien.

```
>v=[]; for i=1 to 10000; v=v|i; end;
```

EMT akan mengkonstruksi sebuah vektor dengan v dan i yang ditambahkan di atas tumpukan (stack) dan menyalin vektor tersebut kembali ke variabel global v.

Cara yang lebih efisien adalah dengan mendefinisikan vektor sebelumnya.

```
>v=zeros(10000); for i=1 to 10000; v[i]=i; end;
```

Untuk mengubah tipe data dalam EMT, Anda dapat menggunakan fungsi seperti complex().

```
>complex(1:4)
```

```
[ 1+0i ,  2+0i ,  3+0i ,  4+0i  ]
```

Konversi ke string hanya mungkin untuk tipe data dasar. Format saat ini digunakan untuk penggabungan string sederhana. Tetapi ada fungsi seperti print() atau frac().

Untuk vektor, Anda dapat dengan mudah menulis fungsi sendiri.

```
>function tostr (v) ...
```

```
  s="[";
  loop 1 to length(v);
     s=s+print(v[#],2,0);
     if #<length(v) then s=s+","; endif;
  end;
  return s+"]";
  endfunction
```

```
>tostr(linspace(0,1,10))
```

```
[0.00,0.10,0.20,0.30,0.40,0.50,0.60,0.70,0.80,0.90,1.00]
```

Untuk berkomunikasi dengan Maxima, ada fungsi convertmxm(), yang juga dapat digunakan untuk memformat vektor untuk keluaran.

```
>convertmxm(1:10)
```

```
[1,2,3,4,5,6,7,8,9,10]
```

Untuk LaTeX, perintah tex dapat digunakan untuk mendapatkan perintah LaTeX.

```
>tex(&[1,2,3])
```

```
\left[ 1 , 2 , 3 \right]
```

Faktor dan Tabel

Dalam pengantar ke R, ada contoh dengan faktor-faktor yang disebutkan.

Berikut adalah daftar wilayah dari 30 negara bagian.

```
>austates = ["tas", "sa", "qld", "nsw", "nsw", "nt", "wa", "wa", ...
>"qld", "vic", "nsw", "vic", "qld", "qld", "sa", "tas", ...
>"sa", "nt", "wa", "vic", "qld", "nsw", "nsw", "wa", ...
>"sa", "act", "nsw", "vic", "vic", "act"];
```

Misalkan, kita memiliki pendapatan yang sesuai di setiap negara bagian.

```
>incomes = [60, 49, 40, 61, 64, 60, 59, 54, 62, 69, 70, 42, 56, ...
>61, 61, 61, 58, 51, 48, 65, 49, 49, 41, 48, 52, 46, ...
>59, 46, 58, 43];
```

Sekarang, kita ingin menghitung rata-rata pendapatan di wilayah-wilayah tersebut. Sebagai program statistik, R memiliki fungsi factor() dan tapply() untuk ini.
EMT dapat melakukan ini dengan menemukan indeks wilayah di daftar unik wilayah.

```
>auterr=sort(unique(austates)); f=indexofsorted(auterr,austates)
```

```
[6,  5,  4,  2,  2,  3,  8,  8,  4,  7,  2,  7,  4,  4,  5,  6,  5,  3,
8,  7,  4,  2,  2,  8,  5,  1,  2,  7,  7,  1]
```

Pada saat itu, kita dapat menulis fungsi loop sendiri untuk melakukan sesuatu hanya untuk satu faktor saja.
Atau kita dapat meniru fungsi tapply() dengan cara berikut.

```
>function map tappl (i; f$:call, cat, x) ...
```

```
u=sort(unique(cat));
f=indexof(u,cat);
return f$(x[nonzeros(f==indexof(u,i))]);
endfunction
```

Ini agak tidak efisien, karena menghitung wilayah-wilayah unik untuk setiap i, tetapi ini berfungsi.

```
>tappl(auterr,"mean",austates,incomes)
```

```
[44.5,  57.3333333333,  55.5,  53.6,  55,  60.5,  56,  52.25]
```

Perlu dicatat bahwa ini berfungsi untuk setiap vektor wilayah.

```
>tappl(["act","nsw"],"mean",austates,incomes)
```

```
[44.5,  57.3333333333]
```

Sekarang, paket statistik EMT mendefinisikan tabel seperti dalam R. Fungsi readtable() dan writetable() dapat digunakan untuk input dan output.

Jadi kita bisa mencetak rata-rata pendapatan negara bagian di wilayah-wilayah dengan cara yang ramah.

```
>writetable(tappl(auterr,"mean",austates,incomes),labc=auterr,wc=7)
```

```
    act    nsw     nt    qld     sa    tas    vic     wa
   44.5  57.33   55.5   53.6     55   60.5     56  52.25
```

Kita juga dapat mencoba meniru perilaku R sepenuhnya.

Faktor-faktor harus jelas disimpan dalam koleksi dengan tipe dan kategori (negara bagian dan wilayah dalam contoh kita). Untuk EMT, kita tambahkan indeks yang telah dihitung sebelumnya.

```
>function makef (t) ...
```

```
  ## Factor data
  ## Returns a collection with data t, unique data, indices.
  ## See: tapply
  u=sort(unique(t));
  return {{t,u,indexofsorted(u,t)}};
  endfunction
```

```
>statef=makef(austates);
```

Sekarang, elemen ketiga dari koleksi akan berisi indeks.

```
>statef[3]
```

```
  [6,  5,  4,  2,  2,  3,  8,  8,  4,  7,  2,  7,  4,  4,  5,  6,  5,  3,
  8,  7,  4,  2,  2,  8,  5,  1,  2,  7,  7,  1]
```

Sekarang kita bisa meniru tapply() dengan cara berikut. Ini akan mengembalikan tabel sebagai koleksi data tabel dan judul kolom.

```
>function tapply (t:vector,tf,f$:call) ...
```

```
## Makes a table of data and factors
## tf : output of makef()
## See: makef
uf=tf[2]; f=tf[3]; x=zeros(length(uf));
for i=1 to length(uf);
   ind=nonzeros(f==i);
   if length(ind)==0 then x[i]=NAN;
   else x[i]=f$(t[ind]);
   endif;
end;
return {{x,uf}};
endfunction
```

Kami tidak menambahkan banyak pemeriksaan tipe di sini. Satu-satunya tindakan pencegahan adalah berkaitan dengan kategori (faktor) tanpa data. Namun, seharusnya memeriksa panjang yang benar dari t dan kebenaran koleksi tf.

Tabel ini dapat dicetak sebagai tabel dengan writetable().

```
>writetable(tapply(incomes,statef,"mean"),wc=7)
```

act	nsw	nt	qld	sa	tas	vic	wa
44.5	57.33	55.5	53.6	55	60.5	56	52.25

Arrays

EMT hanya memiliki dua dimensi untuk array. Tipe data ini disebut matriks. Namun, akan mudah untuk menulis fungsi untuk dimensi yang lebih tinggi atau membuat perpustakaan C untuk ini.

Di R, array adalah vektor dengan bidang dimensi.

Di EMT, sebuah vektor adalah matriks dengan satu baris. Ini dapat diubah menjadi matriks dengan redim().

```
>shortformat; X=redim(1:20,4,5)
```

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20

Pengambilan baris dan kolom, atau sub-matriks, mirip dengan dalam R.

```
>X[,2:3]
```

```
         2           3
         7           8
        12          13
        17          18
```

Namun, di R, Anda dapat mengatur daftar indeks khusus dari vektor ke suatu nilai. Hal yang sama hanya mungkin di EMT dengan menggunakan perulangan.

```
>function setmatrixvalue (M, i, j, v) ...
```

```
  loop 1 to max(length(i),length(j),length(v))
     M[i{#},j{#}] = v{#};
  end;
  endfunction
```

Kami menunjukkan ini untuk menunjukkan bahwa matriks dilewatkan dengan referensi di EMT. Jika Anda tidak ingin mengubah matriks asli M, Anda perlu menyalinnya dalam fungsi.

```
>setmatrixvalue(X,1:3,3:-1:1,0); X,
```

```
         1           2           0           4           5
         6           0           8           9          10
         0          12          13          14          15
        16          17          18          19          20
```

Produk luar (outer product) dalam EMT hanya dapat dilakukan antara vektor. Ini dilakukan secara otomatis karena bahasa matriks. Salah satu vektor harus menjadi vektor kolom dan yang lainnya vektor baris.

```
>(1:5)*(1:5)'
```

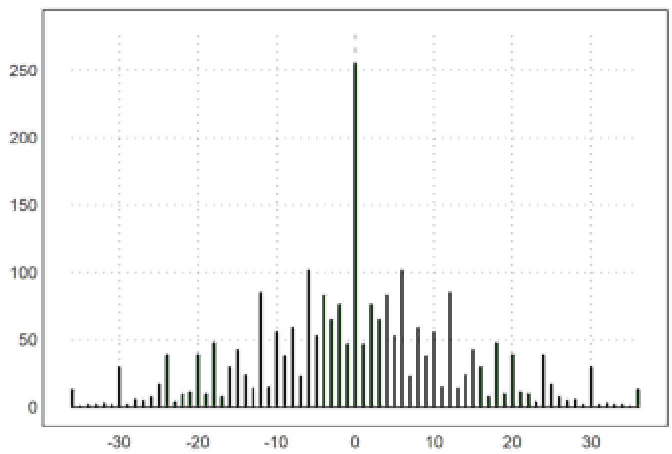
```
         1           2           3           4           5
         2           4           6           8          10
         3           6           9          12          15
         4           8          12          16          20
         5          10          15          20          25
```

Di dalam PDF pengantar untuk R terdapat contoh yang menghitung distribusi ab-cd untuk a, b, c, d yang dipilih dari 0 hingga n secara acak. Solusi dalam R adalah dengan menggunakan matriks 4 dimensi dan menjalankan fungsi table() di atasnya.

Tentu saja, ini dapat dicapai dengan menggunakan perulangan. Namun, perulangan tidak efektif di EMT atau R. Di EMT, kita bisa menulis perulangan dalam bahasa C dan itu akan menjadi solusi yang paling cepat.

Namun, kita ingin meniru perilaku R. Untuk ini, kita perlu meratakan perkalian ab dan membuat matriks ab-cd.

```
>a=0:6; b=a'; p=flatten(a*b); q=flatten(p-p'); ...
>u=sort(unique(q)); f=getmultiplicities(u,q); ...
>statplot(u,f,"h"):
```



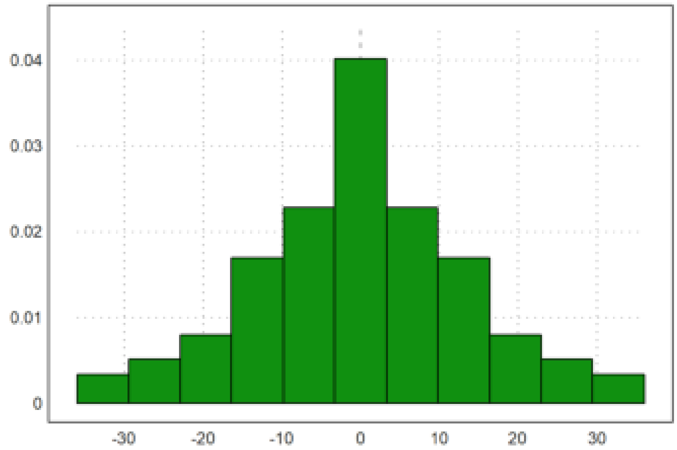
Selain jumlah yang tepat, EMT dapat menghitung frekuensi dalam vektor.

```
>getfrequencies(q,-50:10:50)
```

```
[0,  23,  132,  316,  602,  801,  333,  141,  53,  0]
```

Cara paling mudah untuk menggambarkannya sebagai distribusi adalah sebagai berikut.

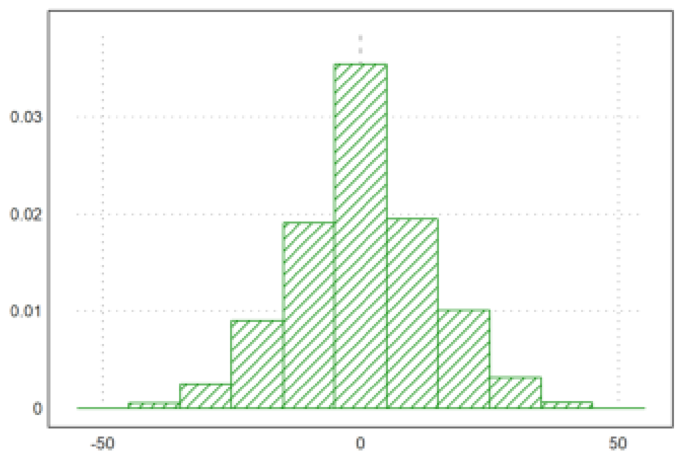
```
>plot2d(q,distribution=11):
```



Tetapi juga mungkin untuk menghitung jumlah dalam interval yang telah dipilih sebelumnya. Tentu saja, berikut ini menggunakan getfrequencies() secara internal.

Karena fungsi histo() mengembalikan frekuensi, kita perlu menyesuaikan sehingga integral di bawah grafik batangnya adalah 1.

```
>{x,y}=histo(q,v=-55:10:55); y=y/sum(y)/differences(x); ...
>plot2d(x,y,>bar,style="/"):
```



Lists

EMT memiliki dua jenis daftar. Satu adalah daftar global yang dapat diubah, dan yang lainnya adalah tipe daftar yang tidak dapat diubah. Kami tidak membahas tentang daftar global di sini.

Tipe daftar yang tidak dapat diubah disebut koleksi dalam EMT. Ini berperilaku seperti struktur dalam C, tetapi elemennya hanya diberi nomor dan tidak dinamai.

```
>L={{"Fred","Flintstone",40,[1990,1992]}}
```

```
Fred
Flintstone
40
[1990,  1992]
```

Saat ini, elemen-elemen tidak memiliki nama, meskipun nama dapat diatur untuk tujuan khusus. Mereka diakses dengan nomor.

```
>(L[4])[2]
```

```
1992
```

Input dan Output Berkas (Membaca dan Menulis Data)

Anda seringkali akan ingin mengimpor matriks data dari sumber lain ke EMT. Panduan ini memberi tahu Anda tentang banyak cara untuk mencapai ini. Fungsi sederhana adalah writematrix() dan readmatrix().

Mari kita tunjukkan bagaimana cara membaca dan menulis vektor bilangan riil ke berkas.

```
>a=random(1,100); mean(a), dev(a),
```

```
0.54004
0.28329
```

Untuk menulis data ke berkas, kita menggunakan fungsi writematrix().

Karena pengantar ini kemungkinan besar berada di direktori di mana pengguna tidak memiliki izin menulis, kita menulis data ke direktori beranda pengguna. Untuk notebook Anda sendiri, hal ini tidak perlu, karena berkas data akan ditulis ke dalam direktori yang sama.

```
>filename="test.dat";
```

Sekarang kita menulis vektor kolom \(\mathbf{a'} \) ke dalam file. Hal ini akan menghasilkan satu angka di setiap baris file tersebut.

```
>writematrix(a',filename);
```

Untuk membaca data tersebut, kita menggunakan fungsi readmatrix().

```
>a=readmatrix(filename)';
```

Dan menghapus file tersebut.

```
>fileremove(filename);
>mean(a), dev(a),
```

```
0.54004
0.28329
```

Fungsi writematrix() atau writetable() dapat dikonfigurasi untuk bahasa lain.

Contohnya, jika Anda menggunakan sistem berbahasa Indonesia (titik desimal digantikan oleh koma), Excel Anda memerlukan nilai dengan koma desimal yang dipisahkan oleh titik koma dalam file csv (yang secara default dipisahkan oleh koma). File berikut "test.csv" akan muncul di folder Anda saat ini.

```
>filename="test.csv"; ...
>writematrix(random(5,3),file=filename,separator=",");
```

Anda sekarang dapat membuka file ini langsung dengan Excel berbahasa Indonesia.

```
>fileremove(filename);
```

Terkadang kita memiliki string dengan token-token seperti contoh berikut.

```
>s1:="f m m f m m m f f f m m f";  ...
>s2:="f f f m m f f";
```

Untuk mengonversi ini menjadi token, kita akan mendefinisikan sebuah vektor dari token-token tersebut.

```
>tok:=["f","m"]
```

```
f
m
```

Kemudian kita dapat menghitung berapa kali setiap token muncul dalam string tersebut, dan memasukkan hasilnya ke dalam sebuah tabel.

```
>M:=getmultiplicities(tok,strtokens(s1))_ ...
>  getmultiplicities(tok,strtokens(s2));
```

Menuliskan tabel dengan judul token.

```
>writetable(M,labc=tok,labr=1:2,wc=8)
```

```
               f       m
       1       6       7
       2       5       2
```

Untuk statistik, EMT dapat membaca dan menulis tabel.

```
>file="test.dat"; open(file,"w"); ...
>writeln("A,B,C"); writematrix(random(3,3)); ...
>close();
```

File tersebut terlihat seperti ini.

```
>printfile(file)
```

```
A,B,C
0.3084163350124789,0.7860228753704319,0.3640480404459329
0.3235527726365187,0.5355164285371529,0.4209486538454125
0.9647906143251817,0.4305557169007305,0.8295841224094156
```

Fungsi readtable() dalam bentuk paling sederhana dapat membaca ini dan mengembalikan kumpulan nilai dan baris judul.

```
>L=readtable(file,>list);
```

Kumpulan ini dapat dicetak menggunakan writetable() ke buku catatan, atau ke dalam sebuah file.

```
>writetable(L,wc=10,dc=5)
```

```
         A         B         C
   0.30842   0.78602   0.36405
   0.32355   0.53552   0.42095
   0.96479   0.43056   0.82958
```

Matriks nilai adalah elemen pertama dari L. Perlu diperhatikan bahwa fungsi mean() dalam EMT menghitung nilai rata-rata dari baris-baris dalam suatu matriks.

```
>mean(L[1])
```

```
0.48616
0.42667
0.74164
```

File CSV

Pertama, mari tulis sebuah matriks ke dalam sebuah file. Untuk output, kita akan membuat file dalam direktori kerja saat ini.

```
>file="test.csv";  ...
>M=random(3,3); writematrix(M,file);
```

Berikut adalah konten dari file tersebut.

```
>printfile(file)
```

```
0.4985755441612197,0.8908902130674888,0.230993153822803
0.5388022720805338,0.03150264484701902,0.9359045715778547
0.6011875635483036,0.1012503400474223,0.4840335655691349
```

File CSV ini dapat dibuka pada sistem berbahasa Inggris di Excel dengan mengklik dua kali. Jika Anda mendapatkan file seperti ini di sistem berbahasa Jerman, Anda perlu mengimpor data ke dalam Excel dengan memperhatikan tanda titik desimal.

Namun, tanda titik desimal adalah format default untuk EMT juga. Anda dapat membaca sebuah matriks dari sebuah file dengan menggunakan fungsi readmatrix().

```
>readmatrix(file)
```

```
0.49858    0.89089    0.23099
 0.5388   0.031503     0.9359
0.60119    0.10125    0.48403
```

Memungkinkan untuk menulis beberapa matriks ke dalam satu file. Perintah open() dapat membuka sebuah file untuk penulisan dengan parameter "w". Nilai default adalah "r" untuk membaca.

```
>open(file,"w"); writematrix(M); writematrix(M'); close();
```

Matriks-matriks tersebut dipisahkan oleh sebuah baris kosong. Untuk membaca matriks-matriks tersebut, buka file dan panggil fungsi readmatrix() beberapa kali.

```
>open(file); A=readmatrix(); B=readmatrix(); A==B, close();
```

```
            1           0           0
            0           1           0
            0           0           1
```

Di Excel atau spreadsheet serupa, Anda dapat mengekspor sebuah matriks sebagai CSV (comma separated values). Di Excel 2007, gunakan "save as" dan "other formats", lalu pilih "CSV". Pastikan bahwa tabel saat ini hanya berisi data yang ingin Anda ekspor.

Berikut adalah contohnya.

```
>printfile("excel-data.csv")
```

```
Could not open the file
excel-data.csv
for reading!
Try "trace errors" to inspect local variables after errors.
printfile:
    open(filename,"r");
```

Seperti yang dapat Anda lihat, sistem Jerman saya menggunakan titik koma sebagai pemisah dan tanda desimal koma. Anda dapat mengubah pengaturan ini dalam pengaturan sistem atau di Excel, tetapi ini tidak diperlukan untuk membaca matriks ke dalam EMT.

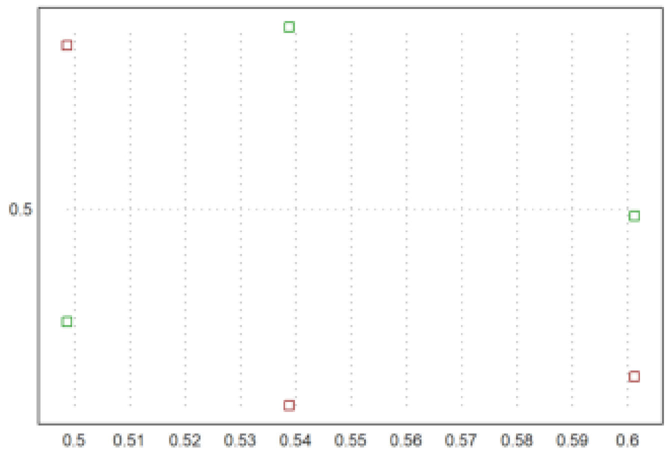
Cara paling mudah untuk membaca ini ke dalam Euler adalah menggunakan readmatrix(). Semua koma akan digantikan oleh titik dengan menggunakan parameter >comma. Untuk CSV berbahasa Inggris, cukup abaikan parameter ini.

```
>M=readmatrix("excel-data.csv",>comma)
```

```
Could not open the file
excel-data.csv
for reading!
Try "trace errors" to inspect local variables after errors.
readmatrix:
    if filename<>"" then open(filename,"r"); endif;
```

Mari memplotnya.

```
>plot2d(M'[1],M'[2:3],>points,color=[red,green]'):
```



Ada cara-cara dasar lain untuk membaca data dari sebuah file. Anda dapat membuka file dan membaca angka-angka satu per satu dari setiap baris. Fungsi getvectorline() akan membaca angka-angka dari sebuah baris data. Secara default, fungsi ini mengharapkan titik desimal. Namun, Anda juga dapat menggunakan koma desimal dengan memanggil setdecimaldot(",") sebelum Anda menggunakan fungsi ini.

Berikut adalah contoh fungsi untuk membaca data tersebut. Fungsi ini akan berhenti pada akhir file atau baris kosong.

```
>function myload (file) ...
```

```
  open(file);
  M=[];
  repeat
     until eof();
     v=getvectorline(3);
     if length(v)>0 then M=M_v; else break; endif;
  end;
  return M;
  close(file);
  endfunction
```

```
>myload(file)
```

```
    0.49858          0    0.89089          0    0.23099
     0.5388          0   0.031503          0     0.9359
    0.60119          0    0.10125          0    0.48403
```

Juga memungkinkan untuk membaca semua angka dalam file tersebut menggunakan fungsi getvector().

```
>open(file); v=getvector(10000); close(); redim(v[1:9],3,3)
```

```
    0.49858          0    0.89089
          0    0.23099     0.5388
          0   0.031503          0
```

Dengan demikian, sangat mudah untuk menyimpan vektor nilai, satu nilai dalam setiap baris, dan membaca kembali vektor ini.

```
>v=random(1000); mean(v)
```

```
  0.49726
```

```
>writematrix(v',file); mean(readmatrix(file)')
```

```
  0.49726
```

Penggunaan Tabel

Tabel dapat digunakan untuk membaca atau menulis data numerik. Sebagai contoh, kita akan menulis sebuah tabel dengan judul baris dan kolom ke dalam sebuah file.

```
>file="test.tab"; M=random(3,3);  ...
>open(file,"w");  ...
>writetable(M,separator=",",labc=["one","two","three"]);  ...
>close(); ...
>printfile(file)
```

```
  one,two,three
        0.58,      0.69,      0.89
        0.88,      0.43,      0.71
        0.18,         1,      0.35
```

Ini dapat diimpor ke dalam Excel.

Untuk membaca file tersebut di EMT, kita menggunakan fungsi readtable().

```
>{M,headings}=readtable(file,>clabs); ...
>writetable(M,labc=headings)
```

```
        one       two     three
       0.58      0.69      0.89
       0.88      0.43      0.71
       0.18         1      0.35
```

Menganalisis Sebuah Baris

Anda bahkan dapat mengevaluasi setiap baris secara manual. Misalkan, kita memiliki sebuah baris dengan format berikut.

```
>line="2020-11-03,Tue,1'114.05"
```

```
2020-11-03,Tue,1'114.05
```

Pertama, kita dapat melakukan tokenisasi terhadap baris tersebut.

```
>vt=strtokens(line)
```

```
2020-11-03
Tue
1'114.05
```

Kemudian, kita dapat mengevaluasi setiap elemen dari baris tersebut menggunakan evaluasi yang sesuai.

```
>day(vt[1]),  ...
>indexof(["mon","tue","wed","thu","fri","sat","sun"],tolower(vt[2])),  ...
>strrepl(vt[3],"'","")()
```

```
7.3816e+05
2
1114
```

Dengan menggunakan regular expressions, mungkin untuk mengekstrak hampir semua informasi dari sebuah baris data.

Misalkan kita memiliki baris berikut dalam sebuah dokumen HTML.

```
>line="<tr><td>1145.45</td><td>5.6</td><td>-4.5</td><tr>"
```

```
<tr><td>1145.45</td><td>5.6</td><td>-4.5</td><tr>
```

Untuk mengekstrak ini, kita menggunakan ekspresi reguler, yang mencari:

- tanda kurung penutup >,
- string apa pun yang tidak mengandung tanda kurung dengan sub-pencocokan "(...)",
- tanda kurung buka dan tanda kurung tutup dengan solusi terpendek,
- lagi string apa pun yang tidak mengandung tanda kurung,
- dan tanda kurung buka <.

Ekspresi reguler agak sulit untuk dipelajari tetapi sangat kuat.

```
>{pos,s,vt}=strxfind(line,">([^<>]+)<.+?>([^<>]+)<");
```

The result is the position of the match, the matched string, and a vector of strings for sub-matches.

```
>for k=1:length(vt); vt[k](), end;
```

```
1145.5
5.6
```

Berikut adalah contoh fungsi yang membaca semua item numerik antara <td> dan </td>.

```
>function readtd (line) ...
```

```
v=[]; cp=0;
repeat
   {pos,s,vt}=strxfind(line,"<td.*?>(.+?)</td>",cp);
   until pos==0;
   if length(vt)>0 then v=v|vt[1]; endif;
   cp=pos+strlen(s);
end;
return v;
endfunction
```

```
>readtd(line+"<td>non-numerical</td>")
```

```
1145.45
5.6
-4.5
non-numerical
```

Membaca dari Web

Sebuah situs web atau file dengan URL dapat dibuka di EMT dan dapat dibaca baris per baris.

Pada contoh ini, kita membaca versi terbaru dari situs EMT. Kita menggunakan ekspresi reguler untuk mencari "Versi ..." dalam judul.

```
>function readversion () ...
```

```
urlopen("http://www.euler-math-toolbox.de/Programs/Changes.html");
repeat
  until urleof();
  s=urlgetline();
  k=strfind(s,"Version ",1);
  if k>0 then substring(s,k,strfind(s,"<",k)-1), break; endif;
end;
urlclose();
endfunction
```

```
>readversion
```

```
Version 2022-05-18
```

Input dan Output Variabel

Anda dapat menulis sebuah variabel dalam bentuk definisi Euler ke sebuah file atau ke baris perintah.

```
>writevar(pi,"mypi");
```

```
mypi = 3.141592653589793;
```

Untuk uji coba, kita akan menghasilkan sebuah file Euler di direktori kerja EMT.

```
>file="test.e"; ...
>writevar(random(2,2),"M",file); ...
>printfile(file,3)
```

```
M = [ ..
0.1715864118118049, 0.7902075714281032;
0.3235267463604796, 0.08344844686478282];
```

Sekarang kita dapat memuat file tersebut. Ini akan mendefinisikan matriks M.

```
>load(file); show M,
```

```
M =
  0.17159   0.79021
  0.32353  0.083448
```

Sebagai informasi tambahan, jika fungsi writevar() digunakan pada sebuah variabel, itu akan mencetak definisi variabel dengan nama variabel tersebut.

```
>writevar(M); writevar(inch$)
```

```
M = [ ..
0.1715864118118049, 0.7902075714281032;
0.3235267463604796, 0.08344844686478282];
inch$ = 0.0254;
```

Kita juga dapat membuka sebuah file baru atau menambahkan ke dalam file yang sudah ada. Pada contoh ini, kita menambahkan ke dalam file yang telah dihasilkan sebelumnya.

```
>open(file,"a"); ...
>writevar(random(2,2),"M1"); ...
>writevar(random(3,1),"M2"); ...
>close();
>load(file); show M1; show M2;
```

```
M1 =
  0.64914   0.13714
    0.112   0.43951
M2 =
  0.77904
  0.52075
  0.88929
```

Untuk menghapus file-file, gunakan fungsi fileremove().

```
>fileremove(file);
```

Sebuah vektor baris dalam sebuah file tidak memerlukan koma jika setiap angka berada di baris baru. Mari kita hasilkan file seperti itu, menulis setiap baris satu per satu dengan menggunakan writeln().

```
>open(file,"w"); writeln("M = ["); ...
>for i=1 to 5; writeln(""+random()); end; ...
>writeln("];"); close(); ...
>printfile(file)
```

```
M = [
0.441463853011
0.602559586157
0.8008250194
0.624852131639
0.53481766277
];
```

```
>load(file); M
```

```
[0.44146,  0.60256,  0.80083,  0.62485,  0.53482]
```

Contoh Soal

Contoh Soal 1

Banyaknya perawat di 6 klinik adalah 3,5,6,4,5, dan 6. Dengan memandang data itu sebagai data populasi, hitunglah nilai rata-rata banyaknya perawat di 6 klinik tersebut!
Penyelesaian:

```
>x=[3,5,6,4,5,6]; mean(x),
```

```
4.8333
```

Jadi, nilai rata-rata banyaknya pegawai di 6 klinik tersebut adalah 4.83

Contoh Soal 2

1. Data berikut menunjukkan tinggi badan dari 20 siswa SMA 2 Surabaya.
Siswa yang tinggi badannya dalam rentang 145-150 sebanyak 1 orang, dalam rentang 151-155 sebanyak 2 orang, dalam rentang 156-160 sebanyak 4 orang, dalam rentang 161-165 sebanyak 3 orang, dalam rentang 166-170 sebanyak 3 orang, dalam rentang 171-175 sebanyak 4 orang, dan dalam rentang 176-180 sebanyak 3 orang.
Tentukan rata-rata tinggi badan dari 20 siswa tersebut!
Penyelesaian:
Menentukan tepi bawah kelas yang terkecil

```
>146-0.5
```

```
145.5
```

Menentukan panjang kelas

```
>(150-146)+1
```

```
5
```

Menentukan tepi atas kelas yang terbesar

```
>180+0.5
```

```
180.5
```

```
>r=145.5:5:180.5; v=[1,2,4,3,3,4,3];
>T:=r[1:7]' | r[2:8]' | v'; writetable(T,labc=["TB","TA","Frek"])
```

```
      TB        TA      Frek
   145.5     150.5         1
   150.5     155.5         2
   155.5     160.5         4
   160.5     165.5         3
   165.5     170.5         3
   170.5     175.5         4
   175.5     180.5         3
```

Menentukan titik tengah

```
>(T[,1]+T[,2])/2 //titik tengah dari tiap interval
```

```
        148
        153
        158
        163
        168
        173
        178
```

```
>t=fold(r,[0.5,0.5])
```

```
 [148,  153,  158,  163,  168,  173,  178]
```

```
>mean(t,v)
```

```
 165.25
```

Jadi, nilai rata-rata tinggi badan dari 20 siswa tersebut adalah 165.25.

Contoh 3

Misalnya kita akan menghitung nilai rata-rata(mean) yang terdapat dalam file "test.dat"

```
>filename="test.dat"; ...
>V=random(3,3); writematrix(V,filename);
>printfile(filename),
```

```
 0.1858294139595237,0.2359542823657924,0.2704760003790539
 0.82636390358253,0.156173368112527,0.9576388301324568
 0.04846986155954107,0.8722870876753714,0.5779228155144673
```

```
>readmatrix(filename)
```

```
0.18583          0   0.23595          0   0.27048
0.82636          0   0.15617          0   0.95764
0.04847          0   0.87229          0   0.57792
```

```
>mean(V),
```

```
0.23075
0.64673
0.49956
```

Contoh 4

Disajikan data urut yaitu 45,48,49,50,52,52,52,53,53,54,54,54,54,54,56,56, 56,56,57,57,58,58,58,58,58,58,58,
62,62,62,63,63,64,64,65,67,68,69,70,70,71,73,74.
Buatlah distribusi frekuensi berdasarkan data diatas!
Penyeleaian:

```
- Menentukan range
  range= nilai maks-nilai min
       = 74-45
       = 29
- Menentukan banyak kelas dengan aturan
  struges.
  = 1+3,3 log n, n banyaknya data
  = 1+3,3 log 48
  = 6,64
  = 7
- Menentukan panjang kelas
```

$$p = \frac{range}{banyakkelas}$$

$$p = \frac{29}{7}$$

$$p = 4.14 = 5$$

Berdasarkan pertimbangan beberapa unsur dalam data urut diatas yaitu nilai minimum 45, nilai maksimum 74, banyak kelas yaitu 7, dan panjang kelas yaitu 5 maka dapat dibuat tabel distribusi frekuensi dengan batas bawah kelas pertama yaitu 43 dan batas atas kelas ketujuh yaitu 77. Sehingga dapat ditentukan tepi bawah kelas pertama yaitu 43-0.5=42.5 dan tepi atas kelas ketujuh yaitu 77+0.5=77.5.

```
>r=42.5:5:77.5; v=[1,6,13,15,6,5,2];
>T:=r[1:7]' | r[2:8]' | v'; writetable(T,labc=["TB","TA","Frek"])
```

```
        TB          TA       Frek
      42.5        47.5          1
      47.5        52.5          6
      52.5        57.5         13
      57.5        62.5         15
      62.5        67.5          6
      67.5        72.5          5
      72.5        77.5          2
```

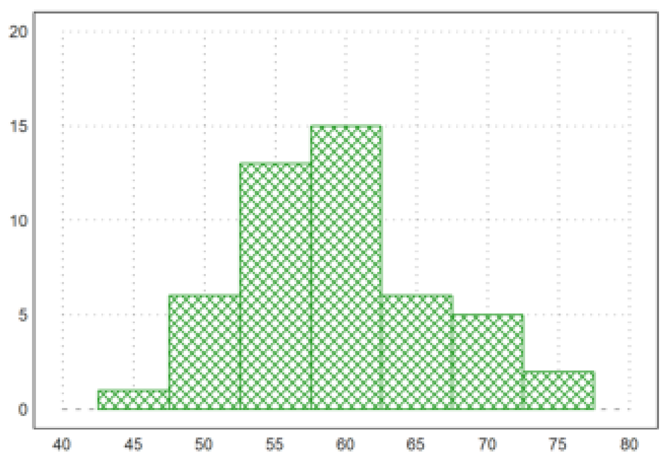
Mencari titik tengah

```
>(T[,1]+T[,2])/2 // titik tengah tiap interval
```

```
        45
        50
        55
        60
        65
        70
        75
```

Sajian dalam bentuk histogram

```
>plot2d(r,v,a=40,b=80,c=0,d=20,bar=1,style="\/"):
```



Contoh 5

Berikut daftar ukuran sepatu anak kelas 5 SDN 1 Makassar.
39,35,35,36,37,38,42,40,38,41,37,35,38,40,41,40.
Tentukan median dari data tersebut!
Penyelesaian :

```
>data=[39,35,35,36,37,38,42,40,38,41,37,35,38,40,41,40];
>urut=sort(data)
```

```
[35,  35,  35,  36,  37,  37,  38,  38,  38,  39,  40,  40,  40,  41,
41,  42]
```

```
>median(data)
```

```
38
```

Jadi, median dari data tersebut adalah 38.

Contoh 6

Berikut adalah data hasil dari pengukuran berat badan 50 siswa SDN 4 Banten. Siswa yang mempunyai berat badan dalam rentang 21-26 kg sebanyak 6 orang, yang mempunyai berat badan dalam rentang 27-32 kg sebanyak 9 orang, yang mempunyai berat badan dalam rentang 33-38 kg sebanyak 14 orang, yang mempunyai berat badan dalam rentang 39-44 kg sebanyak 12 orang, yang mempunyai berat badan dalam rentang 45-50 kg sebanyak 7 orang, dan yang mempunyai berat badan 51-56 kg sebanyak 2 orang. Tentukan median dari data hasil pengukuran berat badan 50 siswa di SD tersebut!
Penyelesaian:
Menentukan tepi bawah kelas yang terkecil

```
>21-0.5
```

```
20.5
```

Menentukan panjang kelas

```
>(26-21)+1
```

```
6
```

Menentukan tepi atas kelas yang terbesar

```
>56+0.5
```

```
56.5
```

```
>r=20.5:6:56.5; v=[6,9,14,12,7,2];
>T:=r[1:6]' | r[2:7]' | v'; writetable(T,labc=["TB","TA","frek"])
```

```
        TB          TA        frek
      20.5        26.5           6
      26.5        32.5           9
      32.5        38.5          14
      38.5        44.5          12
      44.5        50.5           7
      50.5        56.5           2
```

Berdasarkan data, median berada pada urutan ke 25, maka median berada pada kelas 32.5-38.5.

```
>Tb=32.5, p=6, n=50, Fks=15, fm=14
```

```
32.5
6
50
15
14
```

```
>Tb+p*(1/2*n-Fks)/fm
```

```
36.786
```

Jadi, median dari data hasil pengukuran berat badan 50 siswa SDN 4 Banten adalah 36.785.

Contoh 7

Berikut adalah data hasil dari pengukuran berat badan 30 siswa SDN 5 Jember. Siswa yang mempunyai berat badan dalam rentang 21-25 kg sebanyak 1 orang, yang mempunyai berat badan dalam rentang 26-30 kg sebanyak 8 orang, yang mempunyai berat badan dalam rentang 31-35 kg sebanyak 10 orang, yang mempunyai berat badan dalam rentang 36-40 kg sebanyak 5 orang, yang mempunyai berat badan dalam rentang 41-45 kg sebanyak 4 orang, dan yang mempunyai berat badan 46-50 kg sebanyak 2 orang. Tentukan modus dari data hasil pengukuran berat badan 30 siswa di SD tersebut!
Penyelesaian:
Menentukan tepi bawah kelas yang terkecil

```
>21-0.5
```

```
20.5
```

Menentukan panjang kelas

```
>(25-21)+1
```

```
5
```

Menentukan tepi atas yang terbesar

```
>50+0.5
```

```
50.5
```

```
>r=20.5:5:50.5; v=[1,8,10,5,4,2];
>T:=r[1:6]' | r[2:7]' | v'; writetable(T,labc=["TB","TA","frek"])
```

TB	TA	frek
20.5	25.5	1
25.5	30.5	8
30.5	35.5	10
35.5	40.5	5
40.5	45.5	4
45.5	50.5	2

Berdasarkan data, modus berada pada kelas 30.5-35.5.

```
>Tb=30.5, p=5, d1=2, d2=5
```

```
30.5
5
2
5
```

```
>Tb+p*d1/(d1+d2)
```

```
31.929
```

Jadi modus dari data hasil pengukuran berat badan 30 siswa di SDN 5 Jember adalah 31.92.

Contoh 8

Diketahui data sebagai berikut.
11,44,34,51,36,21,23,24,26,27,15,14,16,18,19,20,33,39,45,41,43
Tentukan kuartil, desil, dan persentil dari data tersebut!
Penyelesaian :

```
>data=[11,44,34,51,36,21,23,24,26,27,15,14,16,18,19,20,33,39,45,41,43];
>urut=sort(data)
```

```
[11,  14,  15,  16,  18,  19,  20,  21,  23,  24,  26,  27,  33,  34,
36,  39,  41,  43,  44,  45,  51]
```

```
>quartiles(data)
```

```
[11,  18.5,  26,  40,  51]
```

Dalam output hitung yang dihasilkan dari 'quartiles(data)' dapat diketahui bahwa nilai Q1(kuartil bawah) = 18.5 , Q2(kuartil tengah(median)) = 26, dan Q3(kuartil atas)= 40. Lalu untuk nilai paling kanan dan paling kiri merupakan minimum dan maximum dari suatu data yang diketahui.

```
>quantile(urut,0.1)
```

```
15
```

Dari hasil tersebut dapat diketahui bawha nilai desil ke-1 dan persentil ke-10 adalah 15.

Contoh 9

Diketahui data sebagai berikut.
87,78,88,77,66,67,76
Tentukan varians dan simpangan baku dari data berikut.

```
>data=[87,78,88,77,66,67,76];
>urut=sort(data)
```

```
[66,  67,  76,  77,  78,  87,  88]
```

```
>a=mean(urut)
```

```
77
```

```
>dev=urut-a
```

```
[-11,  -10,  -1,  0,  1,  10,  11]
```

```
>varians=mean(dev^2)
```

```
63.429
```

```
>simpanganBaku= sqrt(varians)
```

```
7.9642
```

Jadi, variansnya adalah 63.42 dan simpangan bakunya adalah 7.96.