November 2024

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^(-2)*y^4*-7*x^2*y^(-7)
```

$$-\frac{42}{y^3}$$

Menjabarkan:

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

```
>$&showev('expand((8*x^(-3)+y^4)*(-7*x^2-y^(-7))))
```

$$expand\left(\left(-\frac{1}{y^7}-7\,x^2\right)\,\left(y^4+\frac{8}{x^3}\right)\right)=-7\,x^2\,y^4-\frac{1}{y^3}-\frac{8}{x^3\,y^7}-\frac{56}{x}$$

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 perintah penugasan atau format.

```
>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 terakhir sebelumnya
```

```
50.2654824574
100.530964915
```

Baris perintah dieksekusi sesuai urutan pengguna menekan tombol enter. Jadi Anda akan mendapatkan nilai baru setiap kali Anda mengeksekusi baris kedua.

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

```
0.540302305868
```

```
>x := cos(x)
```

```
0.857553215846
```

Jika dua baris dihubungkan dengan "..." kedua baris akan selalu dieksekusi secara bersamaan.

```
>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 yang panjang menjadi dua baris atau lebih. Anda dapat menekan Ctrl+Return untuk membagi baris menjadi dua pada posisi kursor saat ini, atau Ctlr+Back untuk menggabungkan baris-baris tersebut.

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

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

Baris yang dimulai dengan %% tidak akan terlihat sama sekali.

```
81
```

Euler mendukung perulangan dalam baris perintah, asalkan dapat dimasukkan ke dalam satu baris atau beberapa baris. Dalam program, pembatasan ini tentu saja tidak berlaku. Untuk informasi lebih lanjut, lihat pengantar berikut.

```
>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
```

Tidak apa-apa menggunakan beberapa baris. Pastikan baris diakhiri 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!
```

Saat Anda menjalankan perintah, kursor dapat berada di posisi mana pun di baris perintah. Anda dapat kembali ke perintah sebelumnya atau melompat ke perintah berikutnya dengan tombol panah. Atau Anda dapat mengklik bagian komentar di atas perintah untuk membuka perintah tersebut.

Saat Anda menggerakkan kursor di sepanjang baris, pasangan tanda kurung buka dan tutup akan disorot. Perhatikan juga baris status. Setelah tanda kurung buka fungsi sqrt(), baris status akan menampilkan teks bantuan untuk fungsi tersebut. Jalankan perintah dengan tombol return.

```
>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 dicari. Pada baris kosong, bantuan untuk jendela bantuan akan ditampilkan. Anda dapat menekan escape untuk menghapus baris, atau untuk menutup jendela bantuan.

Anda dapat mengklik dua kali pada perintah apa pun untuk membuka bantuan untuk perintah ini. Coba klik dua kali perintah exp di bawah ini pada 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 tetikus atau gunakan shift bersamaan dengan tombol kursor apa pun. Selain itu, Anda dapat menyalin tanda kurung yang disorot.

Sintaks Dasar

Euler mengetahui fungsi matematika yang umum. Seperti yang telah Anda lihat di atas, fungsi trigonometri bekerja dalam radian atau derajat. Untuk mengonversi ke derajat, tambahkan simbol derajat (dengan tombol F7) ke nilai, atau gunakan fungsi rad(x). Fungsi akar kuadrat disebut sqrt di Euler. Tentu saja, xˆ(1/2) juga memungkinkan.

Untuk mengatur variabel, gunakan "=" atau ":=". Demi kejelasan, pengantar ini menggunakan bentuk yang terakhir. Spasi tidak menjadi masalah. Namun, spasi di antara perintah diharapkan.

Beberapa perintah dalam satu baris dipisahkan dengan "," atau ";". Titik koma menghilangkan keluaran perintah. Di akhir baris perintah, "," diasumsikan, jika ";" tidak ada.

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

```
30.65625
```

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 menetapkan tanda kurung yang benar dan menggunakan / untuk pecahan. Perhatikan tanda kurung yang disorot untuk mendapatkan bantuan. Perhatikan bahwa konstanta Euler e diberi nama E dalam EMT.

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

```
8.77908249441
```

Untuk menghitung ekspresi rumit seperti

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

Anda perlu memasukkannya dalam formulir baris.

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

```
23.2671801626
```

Letakkan tanda kurung di sekitar sub-ekspresi yang perlu dihitung terlebih dahulu dengan hati-hati. EMT membantu Anda dengan menyorot ekspresi yang diakhiri tanda kurung tutup. Anda juga harus memasukkan nama "pi" untuk huruf Yunani pi.

Hasil perhitungan ini adalah angka floating point. Secara default, angka ini dicetak dengan akurasi sekitar 12 digit. Pada baris perintah berikut, kita juga mempelajari cara merujuk ke hasil sebelumnya dalam baris yang sama.

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

```
0.47619047619
10/21
```

Perintah Euler dapat berupa ekspresi atau perintah primitif. Ekspresi terdiri dari operator dan fungsi. Jika perlu, ekspresi harus berisi tanda kurung untuk memaksakan urutan eksekusi yang benar. Jika ragu, sebaiknya gunakan tanda kurung. Perhatikan bahwa EMT menampilkan tanda kurung buka dan tutup saat mengedit baris perintah.

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

```
14.4978445072
```

Operator numerik Euler meliputi:

```
+ unary or operator plus
- unary or operator minus
*, /
. the matrix product
a^b power for positive a or integer b (a**b works too)
n! the factorial operator
```

and many more.

Berikut ini beberapa fungsi yang mungkin Anda perlukan. Masih banyak lagi.

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

Some commands have aliases, e.g. ln for log.

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

```
2
0.5
```

```
>sin(30°)
```

```
0.5
```

Pastikan untuk menggunakan tanda kurung (kurung bundar), jika ada keraguan tentang urutan eksekusi! Berikut ini tidak sama dengan (2^3)^4, yang merupakan default untuk 2^3^4 dalam EMT (beberapa sistem numerik melakukannya dengan cara lain).

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

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

Contoh Soal

```
>tan(90°)
```

```
1.63312393532e+16
```

```
>10^5*9
```

```
900000
```

Tulis ekspresi yang setara tanpa negatif

```
eksponen dari fungsi berikut :
```

$$(4x^2y)^2$$

```
>$&(4*x^2*y)^2
```

$$16\,x^4\,y^2$$

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

```
>longest 1/3
```

```
      0.3333333333333333
```

Representasi ganda internal membutuhkan 8 byte.

```
>printdual(1/3)
```

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

```
>printhex(1/3)
```

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

String dalam Euler didefinisikan dengan "...".

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

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

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

```
>"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.
```

Fungsi cetak juga mengonversi angka menjadi string. Fungsi ini dapat mengambil sejumlah digit dan sejumlah tempat (0 untuk keluaran padat), dan optimalnya satu unit.

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

```
 Golden Ratio : 1.61803
```

Ada string khusus none, yang tidak dicetak. String ini dikembalikan oleh beberapa fungsi, ketika hasilnya tidak penting. (Dikembalikan secara otomatis, jika fungsi tersebut tidak memiliki pernyataan return.)

```
>none
```

Untuk mengubah 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
```

Vektor string kosong dilambangkan dengan [none]. Vektor string dapat dirangkai.

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

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

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

String Unicode dapat dirangkai seperti string lainnya.

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

```
 = 45°
```

I

Dalam komentar, entitas yang sama seperti , dll. dapat digunakan. Ini mungkin merupakan alternatif cepat untuk Latex. (Rincian lebih lanjut ada di 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")
```

```
[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 string dengan entitas dalam variabel menjadi string Unicode.

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

```
We have =.
```

Dimungkinkan juga untuk menggunakan entitas numerik.

```
>u"&#196;hnliches"
```

```
Ähnliches
```

Nilai Boolean

Nilai Boolean direpresentasikan dengan 1=benar atau 0=salah dalam Euler. String dapat dibandingkan, seperti halnya angka.

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

```
0
1
```

"dan" adalah operator "&&" dan "atau" adalah operator "||", seperti dalam bahasa C. (Kata "dan" dan "atau" hanya dapat digunakan dalam kondisi "jika".)

```
>2<E && E<3
```

```
1
```

Operator Boolean mematuhi aturan bahasa matriks.

```
>(1:10)>5, nonzeros(%)
```

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

Anda dapat menggunakan fungsi nonzeros() untuk mengekstrak 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]
```

Format keluaran default EMT mencetak 12 digit. Untuk memastikan bahwa kita melihat 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 lengkap, gunakan perintah ”longestformat”, atau kami menggunakan operator ”longest” untuk menampilkan hasil dalam format terpanjang.

```
>longest pi
```

```
      3.141592653589793
```

Berikut adalah representasi heksadesimal internal dari angka ganda.

```
>printhex(pi)
```

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

Format keluaran dapat diubah secara permanen dengan perintah format.

```
>format(12,5); 1/3, pi, sin(1)
```

```
      0.33333
      3.14159
      0.84147
```

Format defaultnya adalah(12).

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

```
  0.333333333333
```

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
```

Fungsi ”longestformat” juga mengatur format skalar.

```
>longestformat; pi
```

```
3.141592653589793
```

Sebagai referensi, berikut adalah daftar format output yang paling penting.

format terpendek format pendek format panjang, format terpanjang
format(panjang,digit) format baik(panjang)
fracformat (panjang)
mengubah bentuk

Keakuratan internal EMT adalah sekitar 16 tempat desimal, yang merupakan standar IEEE. Angka disimpan dalam format internal ini.

Namun, format output EMT dapat diatur dengan cara yang fleksibel.

```
>longestformat; pi,
```

```
3.141592653589793
```

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

```
3.14159
```

Standarnya adalah defformat().

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

Ada operator pendek yang hanya mencetak satu nilai. Operator "terpanjang" akan mencetak semua digit angka yang valid.

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

```
4.934802200544679
```

Ada juga operator pendek 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 terwakili secara tepat. Kesalahannya bertambah 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. Demi kenyamanan, output 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
```

String atau nama dapat digunakan untuk menyimpan ekspresi matematika, yang dapat dievaluasi oleh EMT. Untuk ini, gunakan tanda kurung setelah ekspresi. Jika Anda ingin menggunakan string sebagai ekspresi, gunakan konvensi untuk menamainya ”fx” atau ”fxy”, dst. Ekspresi lebih diutamakan daripada fungsi.

Variabel global dapat digunakan dalam evaluasi.

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

```
        12.56637061435917
```

Parameter ditetapkan ke x, y, dan z dalam urutan tersebut. Parameter tambahan dapat ditambahkan menggunakan parameter yang ditetapkan.

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

```
  -0.919535764538
```

Perhatikan bahwa ekspresi akan selalu menggunakan variabel global, bahkan jika ada variabel dalam suatu fungsi dengan nama yang sama. (Jika tidak, evaluasi ekspresi dalam fungsi dapat memberikan 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" selain nilai global, Anda perlu menambahkan "at=value".

```
>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 sebagai 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 dicatat bahwa mendefinisikan 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
```

Berdasarkan konvensi, ekspresi simbolik atau numerik harus diberi nama fx, fxy, dst. Skema penamaan ini tidak boleh digunakan untuk fungsi.

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

$$x^x\,\left(\log x+1\right)$$

Bentuk khusus dari suatu ekspresi memperbolehkan variabel apa pun sebagai parameter tanpa nama untuk evaluasi ekspresi, bukan hanya "x", "y", dst. Untuk ini, awali ekspresi dengan "@(variabel) ...".

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

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

Hal ini memungkinkan untuk memanipulasi ekspresi dalam variabel lain untuk fungsi EMT yang memerlukan 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 halnya 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 ditetapkan.

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

```
-0.425
```

Suatu ekspresi tidak harus simbolis. Hal ini diperlukan, jika ekspresi tersebut memuat fungsi, yang hanya diketahui dalam kernel numerik, bukan dalam Maxima.

EMT mengerjakan matematika simbolis dengan bantuan Maxima. Untuk detailnya, mulailah dengan tutorial berikut, atau telusuri referensi untuk Maxima. Para ahli di Maxima harus memperhatikan bahwa terdapat perbedaan dalam sintaksis antara sintaksis asli Maxima dan sintaksis default ekspresi simbolis dalam EMT.

Matematika simbolis terintegrasi dengan mulus ke dalam Euler dengan &. Setiap ekspresi yang dimulai dengan & adalah ekspresi simbolis. Ekspresi tersebut dievaluasi dan dicetak oleh Maxima.

Pertama-tama, Maxima memiliki aritmatika "tak terbatas" yang dapat menangani 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)
```

$$2481256778$$

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

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

$$2481256778$$

Untuk mempelajari lebih lanjut tentang fungsi tertentu, klik dua kali pada fungsi tersebut. Misalnya, coba klik dua kali pada "&binomial" di baris perintah sebelumnya. Ini akan membuka dokumentasi Maxima sebagaimana disediakan oleh penulis program tersebut.

Anda akan mempelajari bahwa hal berikut juga berfungsi.

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

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

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

Jika Anda ingin mengganti x dengan nilai tertentu, gunakan ”with”.

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


Dengan cara itu Anda dapat menggunakan solusi persamaan dalam persamaan lain.

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

Seperti yang telah Anda lihat pada contoh sebelumnya dan berikutnya, jika Anda telah menginstal LaTeX, Anda dapat mencetak ekspresi simbolik dengan Latex. Jika tidak, perintah berikut akan mengeluarkan 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 menginstal LaTeX.

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

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

Ekspresi simbolik diurai oleh Euler. Jika Anda memerlukan sintaksis yang kompleks dalam satu ekspresi, Anda dapat melampirkan ekspresi tersebut dalam "...". Menggunakan lebih dari satu ekspresi sederhana dimungkinkan, tetapi sangat tidak disarankan.

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


Untuk kelengkapan, kami mencatat bahwa ekspresi simbolik dapat digunakan dalam program, tetapi harus disertakan dalam tanda kutip. Selain itu, akan jauh lebih efektif untuk memanggil Maxima pada waktu kompilasi jika memungkinkan.

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

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

Sekali lagi, % merujuk pada hasil sebelumnya.

Untuk mempermudah, kami menyimpan solusi ke 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. Awali 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
```

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

```
>::: av:g$ av^2;
```

```
 2
g
```

```
>fx &= x^3*exp(x), $fx
```

```
                          3  x
                         x  E
```

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

Variabel tersebut dapat digunakan dalam ekspresi simbolik lainnya. Perhatikan bahwa dalam perintah berikut sisi kanan &= dievaluasi sebelum penugasan ke Fx.

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

```
                              5
                         125 E
```

$$125\,e^5$$

```
                   18551.64488782208
```

```
>fx(5)
```

```
18551.6448878
```

Untuk mengevaluasi ekspresi dengan nilai variabel tertentu, Anda dapat menggunakan operator ”with”.

Baris perintah berikut juga menunjukkan bahwa Maxima dapat mengevaluasi 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 halnya ekspresi numerik.

```
>fx(0.5)
```

```
0.206090158838
```

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

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

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

Penugasan tersebut juga dapat bersifat simbolis.

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

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

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

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

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

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

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

```
1.56155281281
```

Nilai numerik dari solusi simbolik dapat dihitung dengan mengevaluasi hasil simbolik. Euler akan membaca ulang penugasan x= dst. Jika Anda tidak memerlukan hasil numerik untuk perhitungan lebih lanjut, Anda juga dapat membiarkan Maxima menemukan 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 simbolis yang spesifik, seseorang dapat menggunakan "with" dan indeks.

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

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

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

Untuk menyelesaikan 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]
```

$$2$$

Ekspresi simbolik dapat memiliki tanda, yang menunjukkan perlakuan khusus di Maxima. Beberapa tanda dapat digunakan sebagai perintah juga, yang lainnya tidak. Tanda ditambahkan dengan "|" (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(%)
```

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

Fungsi

Dalam EMT, fungsi adalah program yang didefinisikan dengan perintah "function". Fungsi ini dapat berupa fungsi satu baris atau fungsi multibaris.
Fungsi satu baris dapat berupa numerik atau simbolik. Fungsi satu baris numerik didefinisikan oleh ":=".

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

Sebagai gambaran umum, kami tampilkan semua definisi yang mungkin untuk fungsi satu baris. Suatu fungsi dapat dievaluasi seperti fungsi Euler bawaan lainnya.

```
>f(2)
```

```
4.472135955
```

Fungsi ini juga akan bekerja untuk vektor, mematuhi bahasa matriks Euler, karena ekspresi yang digunakan dalam fungsi tersebut divektorkan.

```
>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 menimpa fungsi bawaan, Anda harus menambahkan kata kunci ”overwrite”. Menimpa fungsi bawaan berbahaya dan dapat menyebabkan masalah bagi fungsi lain yang bergantung padanya.

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

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

```
0.707106781187
```

Sebaiknya kita hilangkan pendefinisian ulang dosa ini.

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

```
0.707106781187
```

Fungsi numerik dapat memiliki parameter default.

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

Mengabaikan parameter ini akan menggunakan nilai default.

```
>f(4)
```

```
16
```

Mengaturnya akan menimpa nilai default.

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

```
80
```

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

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

```
16
```

Jika suatu 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 menggantikan nilai global.

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

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

```
20
```

Fungsi simbolik didefinisikan dengan "&=". Fungsi ini didefinisikan dalam Euler dan Maxima, dan berfungsi di kedua dunia. Ekspresi yang 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
```

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

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

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

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

```
178.635099908
```

Mereka dapat digunakan untuk mendefinisikan fungsi atau ekspresi simbolis lainnya.

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

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

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

```
0.703467422498
```

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

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

```
0.703467422498
```

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

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

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

$$(x+2)^n$$

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

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

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

625

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

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

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

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

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

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

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

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

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

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

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

$$x^3-x$$

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

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

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

Dengan := fungsinya adalah numerik. Contoh yang bagus adalah integral tentu seperti

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

yang tidak dapat dievaluasi secara simbolis.

Jika kita mendefinisikan ulang fungsi tersebut dengan kata kunci "map", fungsi tersebut dapat digunakan untuk vektor x. Secara internal, fungsi tersebut dipanggil untuk semua nilai x satu kali, 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.

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

Sekarang fungsi tersebut dapat dipanggil dengan atau tanpa parameter "dasar".

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

```
2
6.7
```

Selain itu, dimungkinkan untuk menggunakan parameter yang ditetapkan.

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

```
2
```

Sering kali, kita ingin menggunakan fungsi untuk vektor di satu tempat, dan untuk elemen individual di tempat lain. Hal ini dimungkinkan dengan parameter vektor.

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

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

Fungsi simbolik semacam itu dapat digunakan untuk variabel simbolik.

Namun, fungsi tersebut juga dapat digunakan untuk vektor numerik.

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

```
17
```

Ada pula 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 kedua
```

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

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

$$0$$

Namun tentu saja, mereka dapat digunakan dalam ekspresi simbolik atau dalam definisi fungsi simbolik.

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

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

Singkatnya

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

Ekspresi dapat diselesaikan secara numerik dan simbolik.

Untuk menyelesaikan ekspresi sederhana dari satu variabel, kita dapat menggunakan fungsi solve(). Fungsi ini memerlukan nilai awal untuk memulai pencarian. Secara internal, solve() menggunakan metode secant.

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

```
1.41421356237
```

Ini juga berlaku untuk ekspresi simbolik. Ambil fungsi berikut.

```
>$&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])
```

$$\left[\left[x=-\frac{c\,e}{b\,\left(d-5\right)-a\,e},y=\frac{c\,\left(d-5\right)}{b\,\left(d-5\right)-a\,e}\right]\right]$$

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

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

Sekarang kita cari titik, di mana polinomialnya adalah 2. Dalam solve(), nilai target default y=0 dapat diubah dengan variabel yang ditetapkan.
Kita gunakan y=2 dan periksa dengan mengevaluasi polinomial pada hasil sebelumnya.

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

```
0.966715594851
2
```

Memecahkan ekspresi simbolik dalam bentuk simbolik akan menghasilkan daftar solusi. Kami menggunakan pemecah simbolik 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 termudah untuk mendapatkan nilai numerik adalah dengan mengevaluasi solusi secara numerik seperti sebuah ekspresi.

```
>longest sol()
```

```
     -0.6180339887498949       1.618033988749895
```

Untuk menggunakan solusi secara simbolis dalam ekspresi lain, cara termudah adalah "dengan".

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

Memecahkan sistem persamaan secara simbolis dapat dilakukan dengan vektor persamaan dan penyelesai simbolis solve(). Jawabannya adalah daftar persamaan.

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

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

Fungsi f() dapat melihat variabel global. Namun, sering kali 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 daftar dengan nama fungsi dan parameter (cara lainnya adalah parameter titik koma).

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

```
2.54116291558
```

Ini juga berlaku untuk ekspresi. Namun, elemen daftar bernama harus digunakan. (Informasi lebih lanjut tentang daftar ada di tutorial tentang sintaks EMT).

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

```
2.54116291558
```

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 pertidaksamaan
```

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

```
>$&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])
```

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

```
>&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])
```

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

Bahasa Matriks

Dokumentasi inti EMT berisi pembahasan terperinci tentang bahasa matriks Euler.

Vektor dan matriks dimasukkan dengan tanda kurung siku, elemen dipisahkan dengan koma, baris dipisahkan dengan titik koma.

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

```
            1             2
            3             4
```

Produk matriks dilambangkan dengan sebuah titik.

```
>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 elemen demi elemen.

```
>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 kolom)
```

```
     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 bukan hasil perkalian matriks, tetapi perkalian elemen demi elemen. Hal yang sama berlaku untuk vektor.

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

```
            9
           16
```

Jika salah satu operan merupakan vektor atau skalar, ia diekspansi dengan cara alami.

```
>2*A
```

```
            2             4
            6             8
```

Misalnya, jika operan adalah vektor kolom, elemen-elemennya diterapkan ke semua baris A.

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

```
            1             4
            3             8
```

Jika itu adalah vektor baris maka diterapkan ke semua kolom A.

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

```
            2             6
            6            12
```

Seseorang dapat membayangkan perkalian ini seolah-olah vektor baris v telah diduplikasi untuk membentuk matriks berukuran sama dengan A.

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

```
          1           2
          1           2
```

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

```
          1           4
          3           8
```

Hal ini juga berlaku untuk dua vektor, yang satu merupakan vektor baris dan yang lainnya merupakan vektor kolom. Kita menghitung i*j untuk i,j dari 1 hingga 5. Caranya 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 bukan produk 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 memperoleh vektor 0 dan 1, di mana 1 berarti benar.

```
>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 tersebut, "nonzeros" memilih elemen yang bukan nol.

Dalam kasus ini, kita memperoleh indeks 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 indeks ini untuk mendapatkan nilai yang sesuai dalam t.

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

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

Sebagai contoh, mari kita cari semua kuadrat angka 1 hingga 1000, yaitu 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 komputasi integer. Ia menggunakan floating point presisi ganda secara internal. Namun, ia sering kali sangat berguna.

Kita dapat memeriksa keutamaan. 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)
```

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

Mengembalikan indeks 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 ini dapat digunakan untuk menetapkan elemen pada 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 pada indeks ke entri 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 adalah mungkin untuk mendapatkan unsur-unsur dalam sebuah vektor.

```
>mget(A,k)
```

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

Fungsi lain yang berguna adalah extrema, yang mengembalikan nilai minimal dan maksimal di setiap baris matriks dan 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 pada 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]
```

Tetapi dengan mget(), kita dapat mengekstrak indeks dan menggunakan informasi ini untuk mengekstrak 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]
```

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

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

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

Dengan cara yang sama, kita dapat menempelkan suatu matriks ke sisi lain yang berdampingan, 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 tidak memiliki jumlah baris yang sama, matriks yang lebih pendek diisi dengan 0.

Ada pengecualian untuk aturan ini. Bilangan riil yang dilampirkan ke matriks akan digunakan sebagai kolom yang diisi dengan bilangan riil 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
```

Dimungkinkan untuk membuat matriks dari vektor baris dan kolom.

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

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

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

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

Tujuan utama dari ini adalah untuk menafsirkan vektor ekspresi untuk vektor kolom.

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

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

Untuk mendapatkan ukuran A, kita dapat menggunakan fungsi berikut.

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

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

Untuk vektor, ada 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 ini.

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

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

Matriks bilangan acak juga dapat dihasilkan dengan acak (distribusi seragam) atau normal (distribusi Gauß).

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

```
        0.66566      0.831835
          0.977      0.544258
```

Berikut adalah fungsi berguna lainnya, yang merestrukturisasi elemen-elemen suatu 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 vektor n kali.

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

Mari kita menguji.

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

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

Fungsi multdup() menduplikasi elemen suatu 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() membalikkan urutan baris atau kolom matriks. Yaitu, fungsi flipx() membalik secara 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 dengan indeks di 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) merujuk pada indeks elemen dalam v, bukan nilai elemen. Jika Anda ingin menghapus elemen, Anda perlu menemukan elemen terlebih dahulu. Fungsi indexof(v,x) dapat digunakan untuk menemukan elemen x dalam vektor v yang 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]
```

Seperti yang Anda lihat, tidak ada salahnya menyertakan indeks di luar rentang (seperti 0), indeks ganda, atau indeks yang tidak diurutkan.

```
>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 membuat 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 atur 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 ini adalah 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 matriks juga dapat diekstraksi dari matriks. Untuk menunjukkan hal ini, kami 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]
```

Misalnya, kita dapat membagi matriks berdasarkan diagonalnya. Bahasa matriks memastikan bahwa vektor kolom d diterapkan ke matriks baris demi baris.

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

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

Hampir semua fungsi di Euler juga berfungsi untuk masukan matriks dan vektor, jika ini masuk akal.

Misalnya, 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 memplot fungsi (alternatifnya menggunakan ekspresi).

```
>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 fungsi dapat dibuat dengan mudah.

Dalam contoh berikut, kita buat vektor nilai t[i] dengan spasi 0,1 dari -1 hingga 1. Kemudian kita buat vektor nilai 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 mengembangkan operator untuk skalar, vektor, dan matriks dengan cara yang jelas.

Misalnya, vektor kolom dikalikan vektor baris akan berkembang menjadi matriks, jika operator diterapkan. Berikut ini, v' adalah vektor yang ditransposisikan (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
```

Perhatikan bahwa ini sangat berbeda dari perkalian matriks. Perkalian matriks dilambangkan dengan titik "." dalam EMT.

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

```
55
```

Secara default, vektor baris dicetak dalam format ringkas.

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

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

Untuk matriks, operator khusus . menunjukkan perkalian matriks, dan A' menunjukkan transposisi. Matriks 1x1 dapat digunakan seperti bilangan riil.

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

```
5
25
```

Untuk mentranspos suatu matriks, kita menggunakan tanda apostrof.

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

```
            1
            2
            3
            4
```

Jadi kita dapat menghitung matriks A dikali vektor b.

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

```
           30
           70
```

Perhatikan bahwa v masih merupakan vektor baris. Jadi v'.v berbeda dengan v.v'.

```
>v'.v
```

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

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

```
>v.v'
```

30

Ada juga fungsi norma (bersama dengan banyak fungsi Aljabar Linier lainnya).

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

30

Operator dan fungsi mematuhi bahasa matriks Euler.

Berikut ringkasan peraturannya.

- Suatu fungsi yang diterapkan pada vektor atau matriks diterapkan pada setiap elemen.

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

- Jika kedua matriks mempunyai dimensi yang berbeda, keduanya diekspansi secara rasional sehingga mempunyai ukuran yang sama.

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

Berikut ini adalah kasus sederhana dengan operator ^.

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

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

Ini kasus yang lebih rumit. Vektor baris dikalikan vektor kolom memperluas keduanya dengan cara menduplikasi.

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

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

Perhatikan bahwa perkalian skalar menggunakan perkalian matriks, bukan *!

```
>v.v'
```

```
14
```

Ada banyak fungsi matriks. Kami memberikan daftar singkat. Anda harus membaca dokumentasi untuk informasi lebih lanjut tentang perintah ini.

```
sum,prod menghitung jumlah dan hasil kali baris
cumsum,cumprod melakukan hal yang sama secara kumulatif
menghitung nilai ekstrem setiap baris
extreme mengembalikan vektor dengan informasi ekstrem
diag(A,i) mengembalikan diagonal ke-i
setdiag(A,i,v) menyetel diagonal ke-i
id(n) matriks identitas
det(A) determinannya
charpoly(A) polinomial karakteristik
nilai eigen(A) nilai eigen
```

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

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

Operator : menghasilkan vektor baris dengan spasi 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 matriks disebut dengan "A[i,j]".

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

```
6
```

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

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

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

Indeks juga dapat berupa vektor baris dari indeks. : menunjukkan semua indeks.

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

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

Bentuk kependekan dari : menghilangkan indeks sepenuhnya.

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

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

Untuk tujuan vektorisasi, elemen matriks dapat diakses seolah-olah elemen tersebut adalah vektor.

```
>A{4}
```

```
4
```

Matriks juga dapat diratakan 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 pada tabel, mari kita atur ulang ke format default, dan hitung tabel nilai sinus dan kosinus. Perhatikan bahwa sudut dinyatakan dalam radian secara default.

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

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

Sekarang kita menambahkan kolom ke 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.

Dalam contoh berikut, kita menghitung t[j]^i untuk i dari 1 hingga n. Kita mendapatkan sebuah matriks, yang setiap barisnya merupakan tabel t^i untuk satu i. Artinya, matriks tersebut memiliki elemen

$$a_{i,j} = t_j^i, \quad 1 \leq j \leq 101, \quad 1 \leq i \leq n$$

Fungsi yang tidak berfungsi untuk masukan vektor harus "divektorkan". Hal ini dapat dicapai dengan kata kunci "peta" dalam definisi fungsi. Kemudian fungsi tersebut akan dievaluasi untuk setiap elemen parameter vektor.

Integrasi numerik integral() hanya berfungsi untuk batas interval skalar. Jadi kita perlu membuat vektorisasinya.

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

Kata kunci "peta" membuat vektorisasi fungsi tersebut. Fungsinya sekarang akan berfungsi untuk vektor bilangan.

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

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

Untuk mengakses elemen matriks, gunakan notasi braket.

```
>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 baris matriks secara lengkap.

```
>A[2]
```

```
 [4,  5,  6]
```

Dalam kasus vektor baris atau kolom, ini mengembalikan elemen vektor.

```
>v=1:3; v[2]
```

```
 2
```

Untuk memastikan, Anda mendapatkan baris pertama untuk matriks 1xn dan mxn, tentukan semua kolom menggunakan indeks kedua yang kosong.

```
>A[2,]
```

```
[4,  5,  6]
```

Jika indeks adalah vektor dari indeks, Euler akan mengembalikan baris matriks yang sesuai.

Di sini kita menginginkan baris pertama dan kedua A.

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

```
             1             2             3
             4             5             6
```

Kita bahkan dapat menyusun ulang A menggunakan vektor indeks. Tepatnya, kita tidak mengubah A di sini, namun menghitung versi A yang disusun ulang.

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

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

Trik indeks juga berfungsi dengan kolom.

Contoh ini memilih semua baris 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
```

Alternatifnya, biarkan indeks pertama kosong.

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

2	3
5	6
8	9

Kita juga bisa mendapatkan baris terakhir A.

```
>A[-1]
```

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

Sekarang mari kita ubah elemen A dengan menetapkan submatriks A ke suatu nilai. Ini sebenarnya mengubah matriks A yang disimpan.

```
>A[1,1]=4
```

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

Kita juga dapat memberikan nilai pada baris A.

```
>A[1]=[-1,-1,-1]
```

```
           -1            -1            -1
            4             5             6
            7             8             9
```

Kita bahkan dapat menetapkan sub-matriks jika ukurannya sesuai.

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

```
              5             6            -1
              7             8             6
              7             8             9
```

Selain itu, beberapa jalan pintas diperbolehkan.

```
>A[1:2,1:2]=0
```

```
              0             0            -1
              0             0             6
              7             8             9
```

Peringatan: Indeks di luar batas mengembalikan matriks kosong, atau pesan kesalahan, bergantung pada pengaturan sistem. Standarnya adalah pesan kesalahan. Namun perlu diingat bahwa indeks negatif dapat digunakan untuk mengakses elemen matriks yang dihitung dari akhir.

```
>A[4]
```

```
Row index 4 out of bounds!
Error in:
A[4] ...
    ^
```

Fungsi sort() mengurutkan vektor baris.

```
>sort([5,6,4,8,1,9])
```

```
[1,  4,  5,  6,  8,  9]
```

Seringkali perlu mengetahui indeks vektor yang diurutkan dalam vektor aslinya. Ini dapat digunakan untuk menyusun ulang vektor lain dengan cara yang sama.

Mari kita mengacak sebuah vektor.

```
>v=shuffle(1:10)
```

```
[4,  5,  10,  6,  8,  9,  1,  7,  2,  3]
```

Indeks berisi urutan 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 entri ganda agak acak.

```
>ind
```

```
[4,  1,  5,  2,  6,  3]
```

Fungsi unik mengembalikan daftar elemen unik vektor yang diurutkan.

```
>intrandom(1,10,10), unique(%)
```

```
[4,  4,  9,  2,  6,  5,  10,  6,  5,  1]
[1,  2,  4,  5,  6,  9,  10]
```

Ini juga berfungsi untuk vektor string.

```
>unique(s)
```

```
a
aa
d
e
```

EMT memiliki banyak sekali fungsi untuk menyelesaikan masalah sistem linier, sistem sparse, atau regresi.

Untuk sistem linier Ax=b, Anda dapat menggunakan algoritma Gauss, matriks invers, atau linear fit. Operator A\b menggunakan versi algoritma Gauss.

```
>A=[1,2;3,4]; b=[5;6]; A\b
```

```
          -4
         4.5
```

Contoh lain, kita membuat matriks berukuran 200x200 dan jumlah baris-barisnya. Kemudian kita selesaikan Ax=b menggunakan matriks invers. Kami mengukur kesalahan sebagai deviasi maksimal semua elemen dari 1, yang tentu saja merupakan solusi yang tepat.

```
>A=normal(200,200); b=sum(A); longest totalmax(abs(inv(A).b-1))
```

```
 8.790745908981989e-13
```

Jika sistem tidak mempunyai solusi, kecocokan linier meminimalkan norma kesalahan Ax-b.

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

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

Penentu matriks ini adalah 0.

```
>det(A)
```

```
0
```

Maxima memiliki matriks simbolik. Tentu saja Maxima dapat digunakan untuk permasalahan aljabar linier sederhana seperti itu. Kita dapat mendefinisikan matriks untuk Euler dan Maxima dengan &:=, lalu menggunakannya dalam ekspresi simbolik. Bentuk [...] yang biasa untuk mendefinisikan matriks dapat digunakan di Euler untuk mendefinisikan matriks simbolik.

```
>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(%)
```

$$(a-1)^2\,(a+2)$$

```
>$&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 simbolik, matriks ini dapat digunakan dalam ekspresi simbolik lainnya.

```
>$&det(A-x*ident(2)), $&solve(%,x)
```

$$\left[x=\frac{3-\sqrt{4\,a\,b+1}}{2} , x=\frac{\sqrt{4\,a\,b+1}+3}{2}\right]$$

$$\left[x=\frac{3-\sqrt{4\,a\,b+1}}{2} , x=\frac{\sqrt{4\,a\,b+1}+3}{2}\right]$$

Nilai eigen juga dapat dihitung secara otomatis. Hasilnya adalah sebuah vektor dengan dua vektor nilai eigen dan multiplisitas.

```
>$&eigenvalues([a,1;1,a])
```

$$\left[\left[a-1 , a+1\right] , \left[1 , 1\right]\right]$$

Untuk mengekstrak vektor eigen tertentu memerlukan pengindeksan yang cermat.

```
>$&eigenvectors([a,1;1,a]), &%[2][1][1]
```

$$[[[a-1,a+1],[1,1]],[[[1,-1]],[[1,1]]]]$$

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

Matriks simbolik dapat dievaluasi dalam Euler secara numerik sama seperti ekspresi simbolik lainnya.

```
>A(a=4,b=5)
```

```
              1             4
              5             2
```

Dalam ekspresi simbolik, gunakan dengan.

```
>$&A with [a=4,b=5]
```

$$\begin{pmatrix}1 & 4\\ 5 & 2\end{pmatrix}$$

Akses ke deretan matriks simbolik berfungsi sama seperti matriks numerik.

```
>$&A[1]
```

$$[1, a]$$

Ekspresi simbolis dapat berisi tugas. Dan itu mengubah matriks A.

```
>&A[1,1]:=t+1; $&A
```

$$\begin{pmatrix} t+1 & a \\ b & 2 \end{pmatrix}$$

Ada fungsi simbolik di 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)
```

$$\begin{pmatrix}-2 & 1 \\ \frac{3}{2} & -\frac{1}{2}\end{pmatrix}$$

$$\begin{pmatrix}-2 & 1 \\ \frac{3}{2} & -\frac{1}{2}\end{pmatrix}$$

Hasilnya dapat dievaluasi secara numerik dalam Euler. Untuk informasi lebih lanjut tentang Maxima, lihat pengenalan Maxima.

```
>$&invert(B)()
```

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

Euler juga memiliki fungsi kuat xinv(), yang melakukan upaya lebih besar dan mendapatkan hasil yang lebih tepat.

Perhatikan, bahwa dengan &:= matriks B telah didefinisikan sebagai simbolik dalam ekspresi simbolik dan numerik dalam ekspresi numerik. Jadi kita bisa menggunakannya di sini.

```
>longest B.xinv(B)
```

```
                      1                         0
                      0                         1
```

Misalnya. 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)
```

$$\left[\left[\frac{15-3\,\sqrt{33}}{2} , \frac{3\,\sqrt{33}+15}{2} , 0 \right] , \left[1 , 1 , 1 \right] \right]$$

Nilai Numerik dalam Ekspresi simbolik

Ekspresi simbolik hanyalah string yang berisi ekspresi. Jika kita ingin mendefinisikan nilai untuk ekspresi simbolik dan ekspresi numerik, kita harus menggunakan "&:=".

```
>A &:= [1,pi;4,5]
```

```
              1         3.14159
              4               5
```

Masih terdapat perbedaan antara bentuk numerik dan simbolik. Saat mentransfer matriks ke bentuk simbolik, pendekatan pecahan untuk real akan digunakan.

```
>$&A
```

$$\begin{pmatrix}1 & \frac{1146408}{364913} \\ 4 & 5\end{pmatrix}$$

Untuk menghindari hal ini, ada fungsi "mxmset(variabel)".

```
>mxmset(A); $&A
```

$$\begin{pmatrix}1 & 3.141592653589793 \\ 4 & 5\end{pmatrix}$$

Maxima juga dapat menghitung dengan bilangan floating point, bahkan dengan bilangan mengambang besar dengan 32 digit. Namun, evaluasinya jauh lebih lambat.

```
>$&bfloat(sqrt(2)), $&float(sqrt(2))
```

$$1.414213562373095$$

Ketepatan angka floating point besar dapat diubah.

```
>&fpprec:=100; &bfloat(pi)
```

```
        3.14159265358979323846264338327950288419716939937510582097494\
 459230781640628620899862803482534211706 8b0
```

Variabel numerik dapat digunakan dalam ekspresi simbolik apa pun menggunakan "@var".

Perhatikan bahwa ini hanya diperlukan, jika variabel telah didefinisikan dengan ":=" atau "=" sebagai variabel numerik.

```
>B:=[1,pi;3,4]; $&det(@B)
```

$$-5.424777960769379$$

Demo - Suku Bunga

Di bawah ini, kami menggunakan Euler Math Toolbox (EMT) untuk menghitung suku bunga. Kami melakukannya secara numerik dan simbolis untuk menunjukkan kepada Anda bagaimana Euler dapat digunakan untuk memecahkan masalah kehidupan nyata.

Asumsikan Anda memiliki modal awal sebesar 5.000 (katakanlah dalam dolar).

```
>K=5000
```

```
5000
```

Sekarang kami mengasumsikan tingkat bunga 3% per tahun. Mari kita tambahkan satu tarif sederhana dan hitung hasilnya.

```
>K*1.03
```

```
5150
```

Euler juga akan memahami sintaks berikut.

```
>K+K*3%
```

```
5150
```

Namun lebih mudah menggunakan faktor tersebut

```
>q=1+3%, K*q
```

```
1.03
5150
```

Selama 10 tahun, kita cukup mengalikan faktor-faktornya dan mendapatkan nilai akhir dengan tingkat bunga majemuk.

```
>K*q^10
```

```
6719.58189672
```

Untuk keperluan kita, kita dapat mengatur format menjadi 2 digit setelah titik desimal.

```
>format(12,2); K*q^10
```

```
    6719.58
```

Mari kita cetak yang dibulatkan menjadi 2 digit dalam satu 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 antara dari tahun 1 sampai tahun ke 9? Untuk ini, bahasa matriks Euler sangat membantu. Anda tidak perlu menulis satu perulangan, tetapi cukup masuk

```
>K*q^(0:10)
```

```
Real 1 x 11 matrix

    5000.00     5150.00     5304.50     5463.64     ...
```

Bagaimana keajaiban ini terjadi? 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 di Euler dapat diterapkan pada vektor elemen demi elemen. 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 q^0 sampai q^10. Ini dikalikan dengan K, dan kita mendapatkan vektor nilainya.

```
>VK=K*q^(0:10);
```

Tentu saja, cara realistis untuk menghitung suku bunga ini adalah dengan membulatkan ke sen terdekat setiap tahunnya. Mari kita tambahkan fungsi untuk ini.

```
>function oneyear (K) := round(K*q,2)
```

Mari kita bandingkan kedua hasil tersebut, 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 mengulanginya selama bertahun-tahun. Euler memberikan banyak solusi untuk ini.

Cara termudah adalah fungsi iterate, yang mengulangi fungsi tertentu beberapa kali.

```
>VKr=iterate("oneyear",5000,10)
```

```
Real 1 x 11 matrix

    5000.00     5150.00     5304.50     5463.64     ...
```

Kami dapat mencetaknya dengan cara yang ramah, menggunakan format kami dengan tempat 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 vektor tertentu, kami menggunakan indeks dalam tanda kurung siku.

```
>VKr[2], VKr[1:3]
```

```
5150.00
5000.00     5150.00     5304.50
```

Anehnya, kita juga bisa menggunakan vektor indeks. Ingatlah bahwa 1:3 menghasilkan vektor [1,2,3]. Mari kita bandingkan elemen terakhir dari nilai yang dibulatkan dengan nilai penuh.

```
>VKr[-1], VK[-1]
```

```
6719.60
6719.58
```

Perbedaannya sangat kecil.

Sekarang kita mengambil fungsi yang lebih maju, yang menambahkan tingkat 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 mendefinisikan nilai-nilai ini. Kami 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     ...
```

Kami melihat uangnya berkurang. Jelasnya, jika kita hanya mendapat bunga sebesar 150 pada tahun pertama, namun menghapus 200, kita kehilangan uang setiap tahunnya.

Bagaimana kita dapat menentukan berapa tahun uang tersebut akan bertahan? Kita harus menulis satu lingkaran untuk ini. Cara termudah adalah dengan melakukan iterasi 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
```

Alasannya adalah bukan nol (VKR<0) mengembalikan vektor indeks i, dengan VKR[i]<0, dan min menghitung indeks minimal.

Karena vektor selalu dimulai dengan indeks 1, maka jawabannya adalah 47 tahun.

Fungsi iterate() memiliki satu trik lagi. Ini dapat mengambil kondisi akhir sebagai argumen. Kemudian akan mengembalikan nilai dan jumlah iterasi.

```
>{x,n}=iterate("onepay",5000,till="x<0"); x, n,
```

```
-19.83
 47.00
```

Mari kita coba menjawab pertanyaan yang lebih ambigu. Asumsikan kita mengetahui bahwa nilainya adalah 0 setelah 50 tahun. Berapa tingkat bunganya?

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 menentukan tingkat suku bunga. Namun untuk saat ini, kami menargetkan solusi numerik.

Langkah pertama adalah mendefinisikan fungsi yang melakukan iterasi sebanyak n kali. Kami 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

$$x_{n+1} = x_n \cdot (1 + \frac{P}{100}) + R$$

Namun kami tidak lagi menggunakan nilai global R dalam ekspresi kami. Fungsi seperti iterate() memiliki trik khusus di Euler. Anda dapat meneruskan nilai variabel dalam ekspresi sebagai parameter titik koma. Dalam hal ini P dan R.

Apalagi kami hanya tertarik pada nilai terakhir. Jadi kita ambil indeks [-1].

Mari kita coba tes.

```
>f(5000,-200,3,47)
```

```
-19.83
```

Sekarang kita bisa menyelesaikan masalah kita.

```
>solve("f(5000,-200,x,50)",3)
```

```
3.15
```

Rutinitas penyelesaian menyelesaikan ekspresi=0 untuk variabel x. Jawabannya adalah 3,15% per tahun. Kami mengambil nilai awal 3% untuk algoritma. Fungsi solve() selalu membutuhkan nilai awal.

Kita dapat menggunakan fungsi yang sama untuk menyelesaikan pertanyaan berikut: Berapa banyak yang dapat kita keluarkan per 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 jumlah tahun, karena fungsi kita mengasumsikan n sebagai nilai bilangan bulat.

Solusi Simbolis Masalah Suku Bunga

Kita dapat menggunakan bagian simbolis dari Euler untuk mempelajari masalahnya. Pertama kita mendefinisikan fungsi onepay() kita secara simbolis.

```
>function op(K) &= K*q+R; $&op(K)
```

$$R + q\,K$$

Sekarang kita dapat mengulanginya.

```
>$&op(op(op(op(K)))), $&expand(%)
```

$$q^3\,R + q^2\,R + q\,R + R + q^4\,K$$

Kami melihat sebuah pola. Setelah n periode yang kita miliki

$$K_n = q^n K + R(1 + q + \ldots + q^{n-1}) = q^n K + \frac{q^n - 1}{q - 1} R$$

Rumusnya adalah rumus jumlah geometri yang diketahui Maxima.

```
>&sum(q^k,k,0,n-1); $& % = ev(%,simpsum)
```

$$\sum_{k=0}^{n-1} q^k = \frac{q^n - 1}{q - 1}$$

Ini agak rumit. Jumlahnya dievaluasi dengan tanda "simpsum" untuk menguranginya menjadi hasil bagi.

Mari kita membuat fungsi untuk ini.

```
>function fs(K,R,P,n) &= (1+P/100)^n*K + ((1+P/100)^n-1)/(P/100)*R; $&fs(K,R,P,n)
```

$$\frac{100\,\left(\left(\frac{P}{100}+1\right)^n-1\right)\,R}{P}+K\,\left(\frac{P}{100}+1\right)^n$$

Fungsinya sama dengan fungsi f kita sebelumnya. Tapi ini lebih efektif.

```
>longest f(5000,-200,3,47), longest fs(5000,-200,3,47)
```

```
-19.82504734650985
-19.82504734652684
```

Sekarang kita dapat menggunakannya untuk menanyakan waktu n. Kapan modal kita habis? Perkiraan awal kami adalah 30 tahun.

```
>solve("fs(5000,-330,3,x)",30)
```

```
20.51
```

Jawaban ini mengatakan akan menjadi negatif setelah 21 tahun.

Kita juga dapat menggunakan sisi simbolis Euler untuk menghitung rumus pembayaran.

Asumsikan kita mendapatkan pinjaman sebesar K, dan membayar n pembayaran sebesar R (dimulai setelah tahun pertama) meninggalkan sisa hutang sebesar Kn (pada saat pembayaran terakhir). Rumusnya 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 diberikan dalam bentuk

$$i = \frac{P}{100}$$

```
>equ &= (equ with P=100*i); $&equ
```

$$\frac{\left(\left(i+1\right)^{n}-1\right)\,R}{i}+\left(i+1\right)^{n}\,K=Kn$$

Kita dapat menyelesaikan nilai R secara simbolis.

```
>$&solve(equ,R)
```

$$\left[R=\frac{i\,Kn-i\,\left(i+1\right)^{n}\,K}{\left(i+1\right)^{n}-1}\right]$$

Seperti yang Anda lihat dari rumusnya, fungsi ini mengembalikan kesalahan floating point untuk i=0. Euler tetap merencanakannya.

Tentu saja, kami memiliki batasan berikut.

```
>$&limit(R(5000,0,x,10),x,0)
```

$$\lim_{x\rightarrow 0}{R\left(5000 , 0 , x , 10\right)}$$

Yang jelas tanpa bunga kita harus membayar kembali 10 tarif 500.

Persamaan tersebut juga dapat diselesaikan untuk n. Akan terlihat lebih bagus jika kita menerapkan beberapa penyederhanaan padanya.

```
>fn &= solve(equ,n) | ratsimp; $&fn
```

$$\left[n=\frac{\log\left(\frac{R+i\,K\,n}{R+i\,K}\right)}{\log\left(i+1\right)}\right]$$

Contoh Latihan Soal

Exercise R.2

1. Sederhanakan bentuk eksponen dari fungsi berikut :

$$\left(\frac{24a^{10}b^{-8}c^{7}}{12a^{6}b^{-3}c^{5}}\right)^{-5}$$

Penyelesaian :

```
>$&((24*a^10*b^(-8)*c^7) / (12*a^6*b^(-3)*c^5))^(-5)
```

$$\frac{b^{25}}{32\,a^{20}\,c^{10}}$$

2. Sederhanakan bentuk fungsi berikut :

$$\left(\frac{125p^{12}q^{-14}r^{22}}{25p^{8}q^{6}r^{-15}}\right)^{-4}$$

Penyelesaian :

```
>$& ((125*p^12*q^-14*r^22) / (25*p^8*q^6*r^-15))^(-4)
```

$$\frac{q^{80}}{625\,p^{16}\,r^{148}}$$

3. Hitunglah operasi di bawah ini :

$$\frac{4(8-6)^2-4\cdot 3+2\cdot 8}{3^1+19^0}$$

Penyelesaian :

```
>&(4*(8-6)^2-4*3+2*8)/(3^1+19^0)
```

5

4. Hitunglah operasi di bawah ini :

$$\frac{[4(8-6)^2+4](3-2\cdot 8)}{2^2(2^3+5)}$$

Penyelesaian :

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

- 5

Exercise Set R.3

6. Selesaikan soal di bawah ini

$$(3x^2 - 2x - x^3 + 2) - (5x^2 - 8x - x^3 + 4)$$

Penyelesaian :

```
>$&(3*x^2-2*x-x^3+2)-(5*x^2-8*x-x^3+4)
```

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

7. Selesaikan soal operasi di bawah ini!

$$(2x + 3y + z - 7) + (4x - 2y - z + 8) + (-3x + y - 2z - 4)$$

Penyelesaian :

```
>$&(2*x+3*y+z-7)+(4*x-2*y-z+8)+(-3*x+y-2*z-4)
```

$$-2\,z + 2\,y + 3\,x - 3$$

8. Selesaikan soal operasi di bawah ini!

$$(2x^2 + 12xy - 11) + (6x^2 - 2x + 4) + (-x^2 - y - 2)$$

Penyelesaian :

```
>$&(2*x^2+12*x*y-11)+(6*x^2-2*x+4)+(-x^2-y-2)
```

$$12\,x\,y - y + 7\,x^2 - 2\,x - 9$$

9. Selesaikan soal operasi di bawah ini!

$$(5x^2 + 4xy - 3y^2 + 2) - (9x^2 - 4xy + 2y^2 - 1)$$

Penyelesaian :

```
>$&(5*x^2+4*x*y-3*y^2+2)-(9*x^2-4*x*y+2*y^2-1)
```

$$-5\,y^2 + 8\,x\,y - 4\,x^2 + 3$$

10. Selesaikan soal operasi di bawah ini!

$$(x^4 - 3x^2 + 4x) - (3x^3 + x^2 - 5x + 3)$$

Penyelesaian :

```
>$&(x^4-3*x^2+4*x)-(3*x^3+x^2-5*x+3)
```

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

11. Asumsikan bahwa semua eksponen adalah bilangan asli. Maka selesaikan soal di bawah ini :

$$(a^n + b^n)(a^n - b^n)$$

Penyelesaian :

```
>$&expand ((a^n+b^n)*(a^n-b^n))
```

$$a^{2\,n} - b^{2\,n}$$

12. Asumsikan bahwa semua eksponen adalah bilangan asli. Maka selesaikan soal di bawah ini:

$$(a+b+c)^2$$

Penyelesaian :

```
>$&expand((a+b+c)^2)
```

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

13. Asumsikan bahwa semua eksponen adalah bilangan asli. Kalikan

$$(x^{3m}-t^{5n})^2$$

Penyelesaian :

```
>$&expand((x^(3*m)-t^(5*n))^2)
```

$$x^{6\,m}-2\,t^{5\,n}\,x^{3\,m}+t^{10\,n}$$

14. Selesaikan soal berikut!

$$t^2 + 8t + 15$$

Penyelesaian :

```
>$&factor(t^2+8*t+15)
```

$$(t+3)\ (t+5)$$

15. Selesaikan soal berikut!

$$y^2 + 12y + 27$$

Penyelesaian :

```
>$&factor(y^2+12*y+27)
```

$$(y+3)\ (y+9)$$

16. Selesaikan soal di bawah ini!

$$x^3 - 4x^2 + 5x - 20$$

Penyelesaian :

```
>$&factor(x^3-4*x^2+5*x-20)
```

$$(x-4)\left(x^2+5\right)$$

17. Selesaikan soal di bawah ini!

$$m^2 - 9n^2$$

Penyelesaian :

```
>$&factor(m^2-9*n^2)
```

$$(m-3\,n)\,(3\,n+m)$$

18. Selesaikan soal di bawah ini!

$$z^3 + 3z^2 - 3z - 9$$

Penyelesaian :

```
>$&factor(z^3+3*z^2-3*z-9)
```

$$(z+3)\left(z^2-3\right)$$

19. Selesaikan soal pefaktoran berikut :

$$18a^2b - 15ab^2$$

Penyelesaian :

```
>$&factor(18*a^2*b-15*a*b^2)
```

$$-3\,a\,b\,(5\,b-6\,a)$$

20. Selesaikan soal pemfaktoran berikut :

$$250z^4 - 2z$$

Penyelesaian :

```
>$&factor(250*z^4-2*z)
```

$$2\,z\,\left(5\,z-1\right)\,\left(25\,z^2+5\,z+1\right)$$

21. Selesaikan soal di bawah ini :

$$m^6 + 8m^3 - 20$$

Penyelesaian :

```
>$&factor(m^6+8*m^3-20)
```

$$\left(m^3-2\right)\,\left(m^3+10\right)$$

22. Selesaikan soal berikut:

$$25ab^4 - 25az^4$$

Penyelesaian :

```
>$&factor(25*a*b^4-25*a*z^4)
```

$$-25\,a\,\left(z-b\right)\,\left(z+b\right)\,\left(z^2+b^2\right)$$

23. Selesaikan soal dibawah ini :

$$16a^7b + 54ab^7$$

Penyelesaian :

```
>$&factor(16*a^7*b+54*a*b^7)
```

$$2\,a\,b\,\left(3\,b^2+2\,a^2\right)\,\left(9\,b^4-6\,a^2\,b^2+4\,a^4\right)$$

24. Selesaikan soal dibawah ini :

$$11x^2 + x^4 - 80$$

Penyelesaian :

```
>$&factor(11*x^2*+x^4-80)
```

$$11\,x^6 - 80$$

25. Selesaikan soal dibawah ini :

$$x^4 - 37x^2 + 36$$

Penyelesaian :

```
>$&factor(x^4-37*x^2+36)
```

$$(x-6)\,(x-1)\,(x+1)\,(x+6)$$

26. Selesaikan soal dibawah ini :

$$2x^2 - 288$$

Penyelesaian :

```
>$&factor(2*x^2-288)
```

$$2\ (x - 12)\ (x + 12)$$

27. Selesaikan soal dibawah ini :

$$(x + 0.001)^2 - x^2$$

Penyelesaian :

```
>$&factor(x+0.001)^2-x^2
```

$$\frac{(1000\,x + 1)^2}{1000000} - x^2$$

28. Selesaikan soal dibawah ini :

$$(x + h)^3 - x^3$$

Penyelesaian :

```
>$&factor((x+h)^3-(x^3))
```

$$h\,\left(3\,x^2+3\,h\,x+h^2\right)$$

Exercise Set R.5

29. Selesaikan soal dibawah ini :

$$7(3x + 6) = 11 - (x + 2)$$

Penyelesaian :

```
>$&solve(7*(3*x+6)=11-(x+2))
```

$$\left[x = -\frac{3}{2}\right]$$

30. Selesaikan soal dibawah ini :

$$9(2x + 8) = 20 - (x + 5)$$

Penyelesaian :

```
>$&solve(9*(2*x+8)=20-(x+5))
```

$$[x = -3]$$

31. Selesaikan soal dibawah ini :

$$4(3y - 1) - 6 = 5(y + 2)$$

Penyelesaian :

```
>$&solve(4*(2*y-1)-6=5*(y+2))
```

$$\left[y = \frac{20}{3}\right]$$

32. Selesaikan soal dibawah ini :

$$3(2n-5)-7=4(n-9)$$

Penyelesaian :

```
>$&solve(3*(2*n-5)-7=4*(n-9))
```

$$[n = -7]$$

Exercise Set R.6

33. Selesaikan soal dibawah ini :

$$\left[\frac{x+1}{x-1}+1\right]^5$$

Penyelesaian :

```
>$&solve(((x+1)/(x-1))+1)^5
```

$$[x^5 = 0]$$

34. Selesaikan soal dibawah ini :

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

Penyelesaian :

```
>$&solve((x^2-4)/(x^2-4*x+4))
```

$$[x = -2]$$

35. Selesaikan soal dibawah ini :

$$\left[\frac{-5}{3}\right]$$

Penyelesaian :

```
>$&solve((-5)/(3))
```

$$[]$$

36. Selesaikan soal dibawah ini :

$$\left[\frac{4}{7-x}\right]$$

Penyelesaian :

```
>$&solve((4)/(7-x))
```

$$[]$$

Exercise Set 2.3

37. Selesaikan soal dibawah ini :

$$(f \circ g)(-1)$$

Penyelesaian :

Soal Bilangan Asli

38. Tuliskan bilangan asli dari

$$\frac{1}{8}$$

Penyelesaian :

```
>longest 1/8
```

```
                    0.125
```

```
>printdual (1/8)
```

```
  1.0000000000000000000000000000000000000000000000000000*2^-3
```

```
>printhex (1/8)
```

```
  2.0000000000000*16^-1
```

39. Tuliskan bilangan asli dari

$$\frac{13}{23}$$

Penyelesaian :

```
>longest 13/23
```

```
      0.5652173913043478
```

```
>printdual (13/23)
```

```
 1.0010000010110010000010110010000010110010000010110010000 1*2^-1
```

```
>printhex (13/23)
```

```
 9.0B21642C85908*16^-1
```

Soal String

40. Tuliskan string dari "Aku suka mata kuliah Aplikom"

Penyelesaian :

```
>"Aku suka mata kuliah Aplikom"
```

```
Aku suka mata kuliah Aplikom
```

Soal Nilai Boolean

41. Apakah benar bahwa:

$$5 > 3 and 2 < 1$$

Penyelesaian :

```
>5 > 3 && 2 < 1
```

```
0.00
```

42. Tulisan format keluaran dari nilai :

$$tan(180)$$

Penyelesaian :

```
>defformat; tan(180)
```

```
1.33869021035
```

```
>longest tan(180)
```

```
      1.338690210351154
```

```
>printhex (tan(180))
```

```
1.56B466D0EEFE6*16^0
```

```
>longestformat; tan(180)
```

```
1.338690210351154
```

Soal Matematika Simbolik

43. Hitunglah hasil dari soal berikut :

$$C(98,10) = \frac{70!}{56! \cdot 10!}$$

Penyelesaian :

```
>$& 70!/(56!*10!) // nilai C(98,10)
```

$$464272825598565 9840$$

```
>$binomial(98,10) //menghitung C(98,10) menggunakan fungsi binomial()
```

$$14005614014756$$

44. Hitunglah hasil dari soal berikut :

$$C(x,9) = \frac{x!}{(x-8)!7!} = \frac{(x-4)(x-2)x}{8}$$

Penyelesaian :

```
>$binomial(x,9) // c(x,9)
```

$$\frac{(x-8)\ (x-7)\ (x-6)\ (x-5)\ (x-4)\ (x-3)\ (x-2)\ (x-1)\ x}{362880}$$

```
>$&binomial(x,9) with x=10 // substitusi x=10 ke C(x,9)
```

$$10$$

```
>&"v := 8; v^2"
```

45. Hitunglah faktor dari 90

Penuyelesaian :

```
>&factor(90!)
```

```
         14857159644817614973095227336208257378855699612846887669422168\
63704985393094065876545992131370884059645617234469978112000000000000\
000000000
```

```
>::: factor(90!)
```

$$2^{86}\,3^{44}\,5^{21}\,7^{13}\,11^{8}\,13^{6}\,17^{5}\,19^{4}\,23^{3}\,29^{3}\,31^{2}\,37^{2}\,41^{2}\,43^{2}\,47\,53\,59\,61\,67\,71\,73\,79\,83\,89$$

46.Tentukan hasil dari fungsi berikut :

$$x^8 + 6$$

Penyelesaian :

Soal Matriks

46. Hitunglah

$$A = [5, 6; 7, 8]$$

Penyelesaian :

```
>A=[5,6;7,8]
```

5	6
7	8

```
>a=[5;6]
```

```
5
6
```

```
>a'
```

```
[5,  6]
```

```
>inv(A)
```

```
-4.000000000000002       3.000000000000001
 3.500000000000002      -2.500000000000001
```

```
>A.a
```

```
61
83
```

```
>A^2
```

```
25      36
49      64
```

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

11587	13494
15743	18334

```
>power(A,8)
```

346694611	403753974
471046303	548571598

```
>A/A
```

1	1
1	1

```
>A/a
```

1	1.2
1.166666666666667	1.333333333333333

```
>A\a
```

```
  -2.000000000000002
   2.500000000000002
```

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

```
                  -2
                 2.5
```

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

```
                   1                       0
                   0                       1
```

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

```
   1.000000000000004                       0
-3.552713678800501e-15                     1
```

```
>A*A
```

25	36
49	64

```
>a^2
```

25
36

```
>A|1
```

```
Real 2 x 3 matrix

            5     ...
            7     ...
```

Soal Masalah suku bunga

47. Sebuah motor bekas akan dijual dengan harga \$78,000 dengan uang muka sebanyak \$12,000. Jika seseorang meminjam uang untuk membeli motor tersebut dengan lama pinjaman 20 tahun dan suku bunga 5,4%, maka berapa jumlah yang harus dibayarkan oleh peminjam di setiap bulannya

Penyelesaian :

```
>K=78000-12000, q=1+5.4%, format(12,2); (K*q^20)/(12*20)
```

```
66000
1.054
     787.31
```

48. Sebuah tanah pekarangan akan dijual dengan harga \$125,000 dengan uang muka sebanyak \$100,000. Jika seseorang meminjam uang untuk membeli tanah pekarangan tersebut dengan lama pinjaman 35 tahun dan suku bunga 8,4%, maka berapa jumlah yang harus dibayarkan oleh peminjam di setiap bulannya

Penyelesaian :

```
>K=125000-100000, q=1+8.4%, format(12,2); (K*q^35)/(12*35)
```

```
   25000.00
       1.08
    1001.65
```

49. Sebuah rumah akan dijual dengan harga $72,000 dengan uang muka sebanyak $50,000. Jika seseorang meminjam uang untuk membeli rumah tersebut dengan lama pinjaman 15 tahun dan suku bunga 6,4%, maka berapa jumlah yang harus dibayarkan oleh peminjam di setiap bulannya

Penyelesaian :

```
>K=72000-50000, q=1+6.4%, format(12,2); (K*q^15)/(12*15)
```

```
    22000.00
        1.06
      309.94
```

50. Sebuah laptop bekas akan dijual dengan harga $45,000 dengan uang muka sebanyak $30,000. Jika seseorang meminjam uang untuk membeli laptop bekas tersebut dengan lama pinjaman 5 tahun dan suku bunga 2,4%, maka berapa jumlah yang harus dibayarkan oleh peminjam di setiap bulannya

Penyelesaian :

```
>K=45000-30000, q=1+2.4%, format(12,2); (K*q^5)/(12*5)
```

```
    15000.00
        1.02
      281.47
```

Nama : Aulia Agnes Luistin
NIM : 23030630001
Kelas : Matematika B

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 antara 0 hingga 1024 di setiap sumbu, tidak peduli apakah layarnya berbentuk persegi atau tidak. Semut terdapat koordinat plot yang dapat diatur dengan setplot(). Pemetaan antar koordinat bergantung pada jendela plot saat ini. Misalnya, shrinkwindow() default menyisakan ruang untuk label sumbu dan judul plot.

Dalam contoh ini, kita hanya menggambar beberapa garis acak dengan berbagai warna. Untuk rincian tentang fungsi-fungsi ini, pelajari fungsi inti EMT.

```
>clg; // untuk membersihkan layar
>window(0,0,1024,1024); // gunakan semua window
>setplot(0,1,0,1); // koordinat set plot
>hold on; // untuk memulai overwrite mode
>n=100; X=random(n,2); Y=random(n,2);  // untuk membuat koordinat acak
>colors=rgb(random(n),random(n),random(n)); // get random colors
>loop 1 to n; color(colors[#]); plot(X[#],Y[#]); end; // plot
>hold off; // mengakhiri overwrite mode
>insimg; //memasukkan ke notebook
```

```
>reset;
```

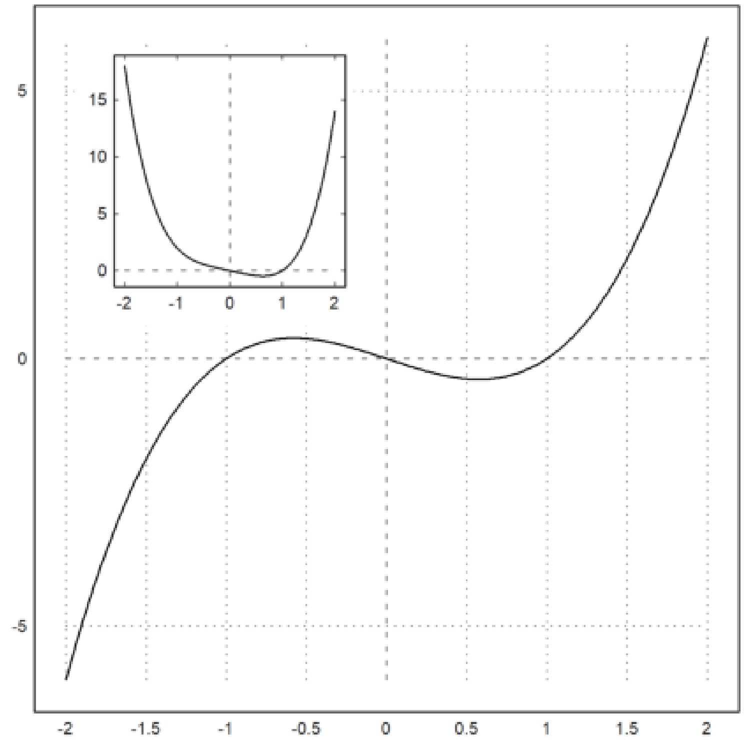
Grafik perlu ditahan, karena perintah plot() akan menghapus jendela plot.

Untuk menghapus semua yang kami lakukan, kami menggunakan reset().

Untuk menampilkan gambar hasil plot di layar notebook, perintah plot2d() dapat diakhiri dengan titik dua (:). Cara lainnya adalah perintah plot2d() diakhiri dengan titik koma (;), kemudian menggunakan perintah insimg() untuk menampilkan gambar hasil plot.

Contoh lain, kita menggambar plot sebagai sisipan di plot lain. Hal ini dilakukan dengan mendefinisikan jendela plot yang lebih kecil. Perhatikan bahwa jendela ini tidak memberikan ruang untuk label sumbu di luar jendela plot. Kita harus menambahkan beberapa margin untuk ini sesuai kebutuhan. Perhatikan bahwa kita menyimpan dan memulihkan jendela penuh, dan menahan plot saat ini sementara kita memplot inset.

```
>plot2d("x^3-x");
>xw=200; yw=100; ww=300; hw=300;
>ow=window();
>window(xw,yw,xw+ww,yw+hw);
>hold on;
>barclear(xw-50,yw-10,ww+60,ww+60);
>plot2d("x^4-x",grid=6):
```




```
>hold off;
>window("x^4-x",grid=6):...
```

```
Argument grid not in parameter list of function window.

Parameters:
```

Plot dengan banyak gambar dicapai dengan cara yang sama. Ada fungsi utilitas figure() untuk ini.

Aspek Plot

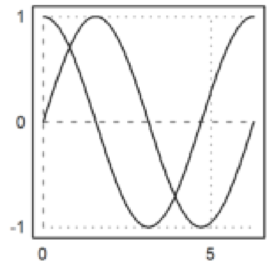
Plot default menggunakan jendela plot persegi. Anda dapat mengubahnya dengan fungsi aspek(). Jangan lupa untuk mengatur ulang aspeknya nanti. Anda juga dapat mengubah default ini di menu dengan "Set Aspect" ke rasio aspek tertentu atau ke ukuran jendela grafik saat ini.

Tapi Anda juga bisa mengubahnya untuk satu plot. Untuk ini, ukuran area plot saat ini diubah, dan jendela diatur sehingga label memiliki cukup ruang.

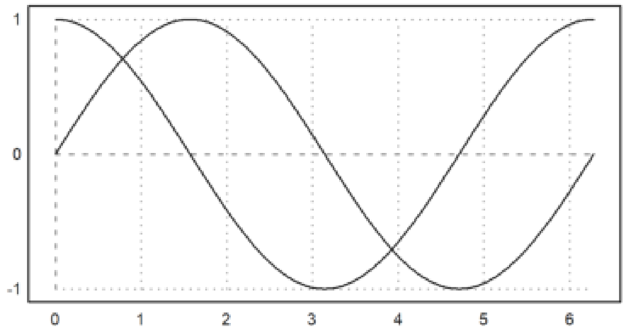
```
>aspect(2); // rasio panjang dan lebar 2:1
```

```
Error in:
aspect(2); // rasio panjang dan lebar 2:1 ...
      ^
```

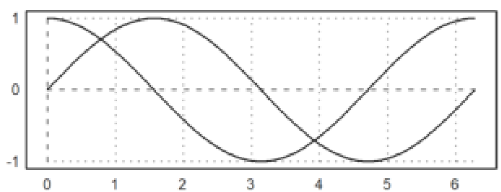
```
>plot2d(["sin(x)","cos(x)"],0,2pi):
```




```
>aspect(2); // rasio panjang dan lebar 2:1
>plot2d(["sin(x)","cos(x)"],0,2pi):
```




```
>aspect(3); // rasio panjang dan lebar 2:1
>plot2d(["sin(x)","cos(x)"],0,2pi):
```




```
>aspect();
>reset;
```

Fungsi reset() mengembalikan default plot termasuk rasio aspek.
Plot 2D di Euler

EMT Math Toolbox memiliki plot dalam 2D, baik untuk data maupun fungsi. EMT menggunakan fungsi plot2d. Fungsi ini dapat memplot fungsi dan data.

Dimungkinkan untuk membuat plot di Maxima menggunakan Gnuplot atau dengan Python menggunakan Math Plot Lib.

Euler dapat membuat plot 2D

- ekspresi
- fungsi, variabel, atau kurva berparameter,
- vektor nilai x-y,
- awan titik di pesawat,
- kurva implisit dengan level atau wilayah level.
- Fungsi kompleks

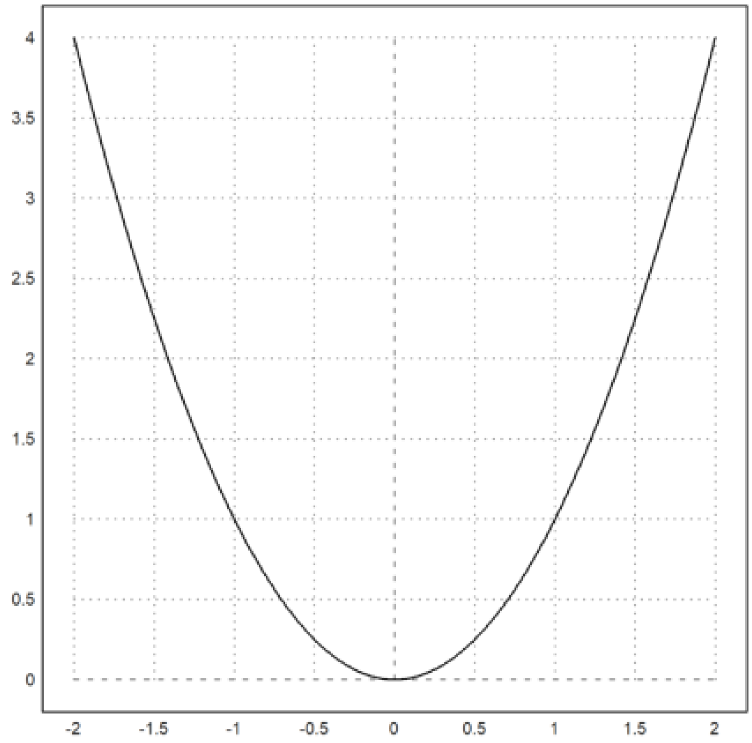
Gaya plot mencakup berbagai gaya untuk garis dan titik, plot batang, dan plot berbayang.
Plot Ekspresi atau Variabel

Ekspresi tunggal dalam ”x” (misalnya ”4*x^2”) atau nama suatu fungsi (misalnya ”f”) menghasilkan grafik fungsi tersebut.

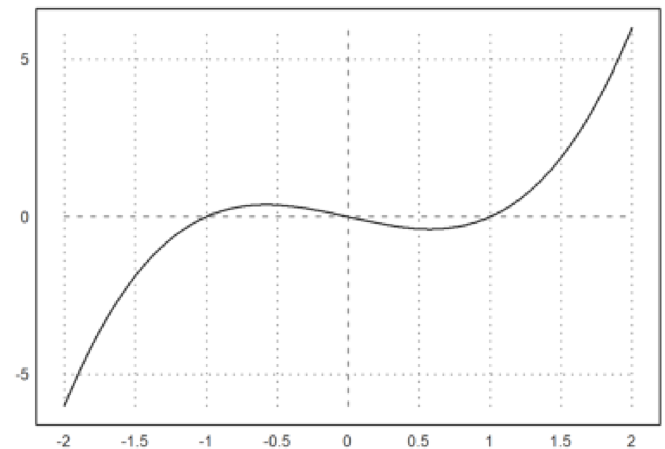
Berikut adalah contoh paling dasar, yang menggunakan rentang default dan menetapkan rentang y yang tepat agar sesuai dengan plot fungsinya.

Catatan: Jika Anda mengakhiri baris perintah dengan titik dua ”:”, plot akan dimasukkan ke dalam jendela teks. Jika tidak, tekan TAB untuk melihat plot jika jendela plot tertutup.

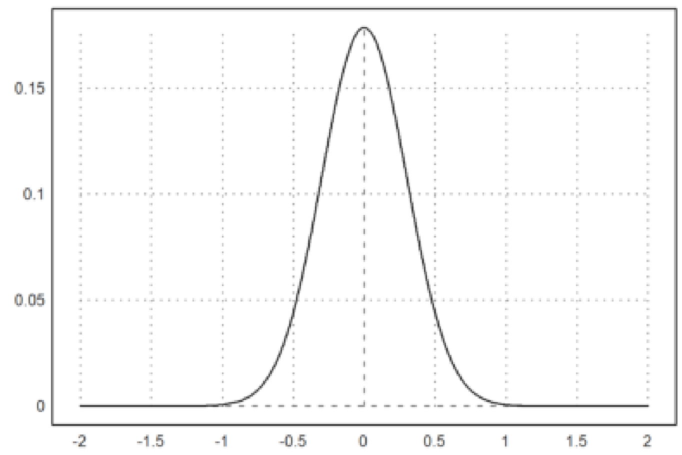
```
>plot2d("x^2"):
```




```
>aspect(1.5); plot2d("x^3-x"):
```




```
>a:=5.6; plot2d("exp(-a*x^2)/a"); insimg(30); // menampilkan gambar hasil plot setinggi 25 baris
```

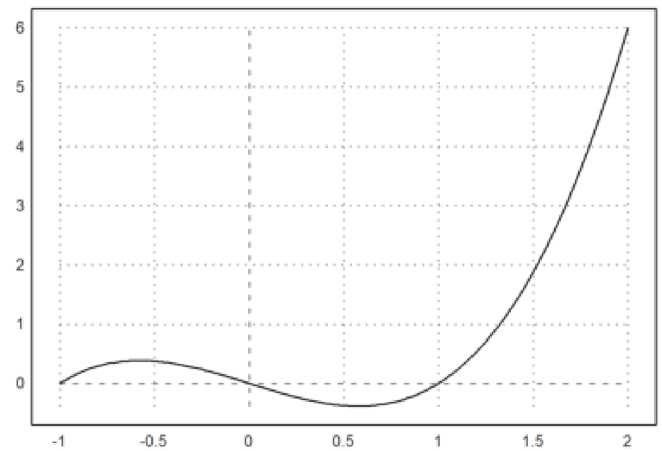


Dari beberapa contoh sebelumnya Anda dapat melihat bahwa aslinya gambar plot menggunakan sumbu X dengan rentang nilai dari -2 sampai dengan 2. Untuk mengubah rentang nilai X dan Y, Anda dapat menambahkan nilai-nilai batas X (dan Y) di belakang ekspresi yang digambar.

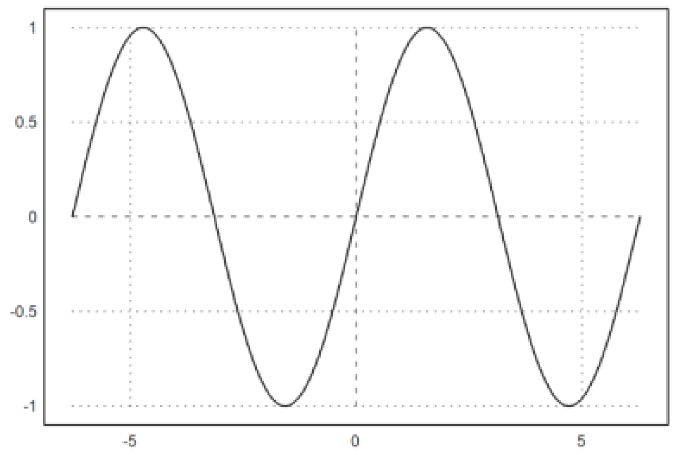
Rentang plot diatur dengan parameter yang ditetapkan sebagai berikut

- a,b: rentang x (default -2,2)
- c,d: rentang y (default: skala dengan nilai)
- r: alternatifnya radius di sekitar pusat plot
- cx,cy: koordinat pusat plot (default 0,0)

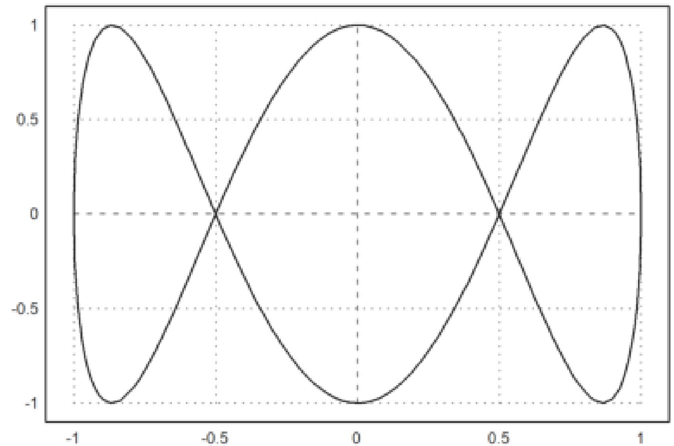
```
>plot2d("x^3-x",-1,2):
```




```
>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 titik dua adalah perintah insimg(baris), yang menyisipkan plot yang menempati sejumlah baris teks tertentu.

Dalam opsi, plot dapat diatur agar muncul

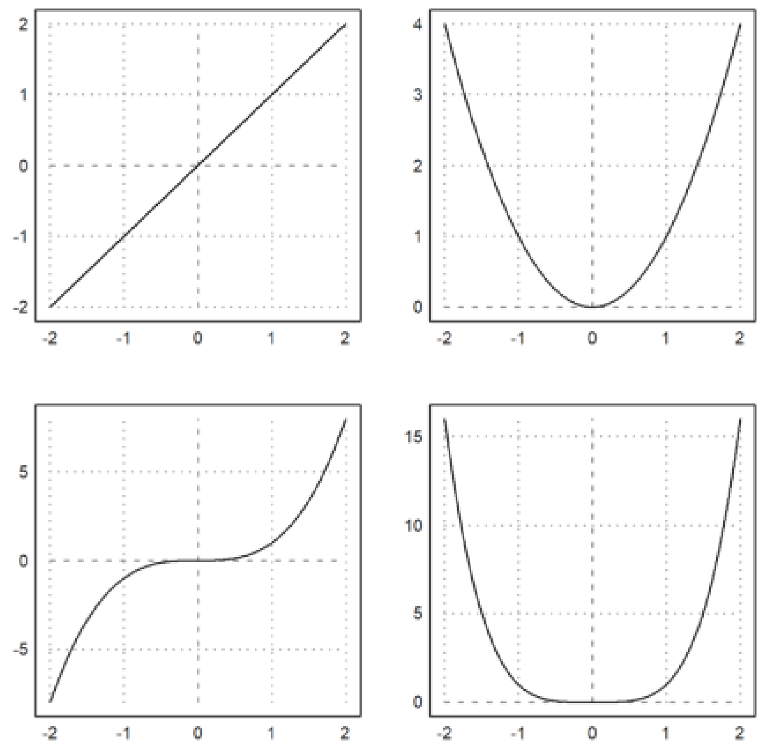
- di jendela terpisah yang dapat diubah ukurannya,
- di jendela buku catatan.

Lebih banyak gaya dapat dicapai dengan perintah plot tertentu.

Bagaimanapun, tekan tombol tabulator untuk melihat plotnya, jika tersembunyi.

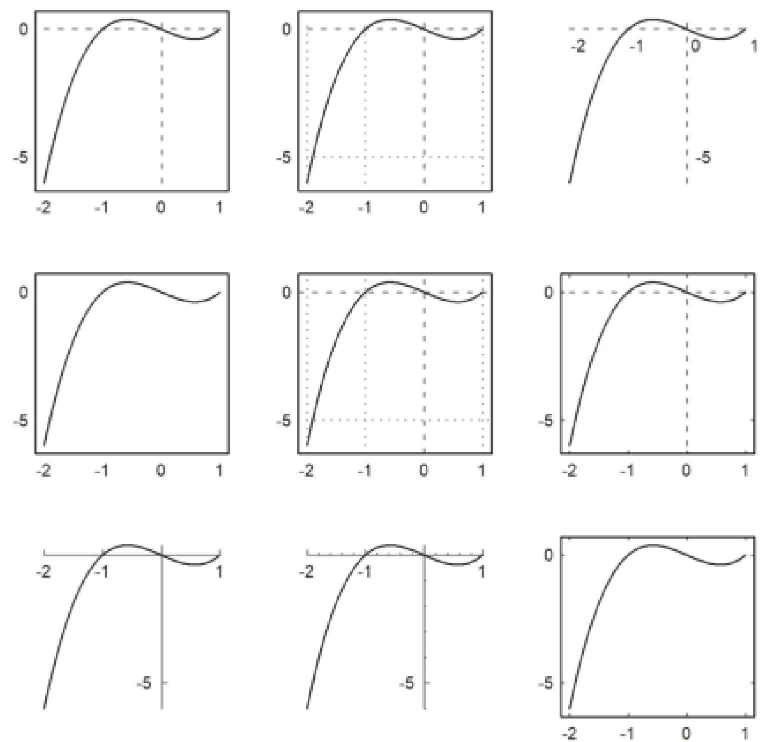
Untuk membagi jendela menjadi beberapa plot, gunakan perintah figure(). Dalam contoh, kita memplot xˆ1 hingga xˆ4 menjadi 4 bagian jendela. gambar(0) mengatur ulang jendela default.

```
>reset;
>figure(2,2); ...
>for n=1 to 4; figure(n); plot2d("x^"+n); end; ...
>figure(0):
```



Di plot2d(), ada gaya alternatif yang tersedia dengan grid=x. Untuk gambaran umum, kami menampilkan berbagai gaya kisi dalam satu gambar (lihat di bawah untuk perintah figure()). Gaya grid=0 tidak disertakan. Ini tidak menunjukkan kisi dan bingkai.

```
>figure(3,3); ...
>for k=1:9; figure(k); plot2d("x^3-x",-2,1,grid=k); end; ...
>figure(0):
```

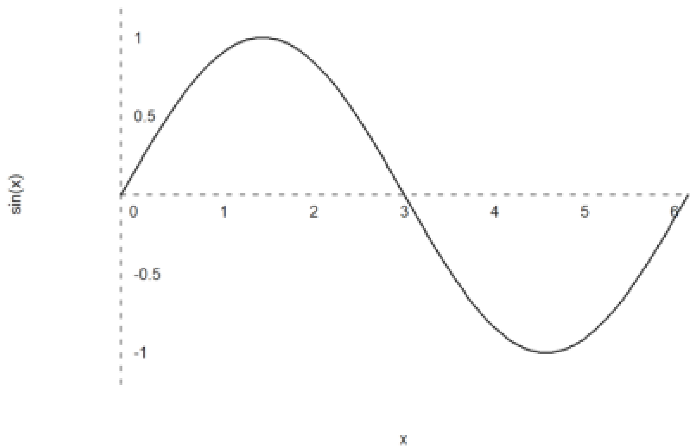


Jika argumen pada plot2d() adalah ekspresi yang diikuti oleh empat angka, angka-angka tersebut adalah rentang x dan y untuk plot tersebut.

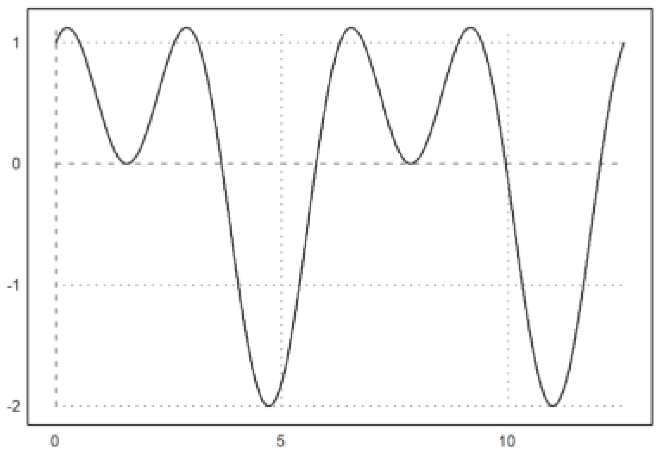
Alternatifnya, a, b, c, d dapat ditentukan sebagai parameter yang ditetapkan sebagai a=... dll.

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):
```

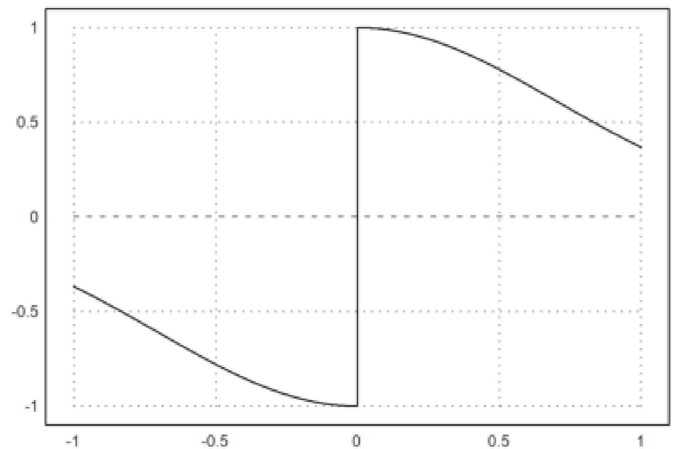


Gambar yang dihasilkan dengan memasukkan plot ke dalam jendela teks disimpan di direktori yang sama dengan buku catatan, secara default di subdirektori bernama ”gambar”. Mereka juga digunakan oleh ekspor HTML.

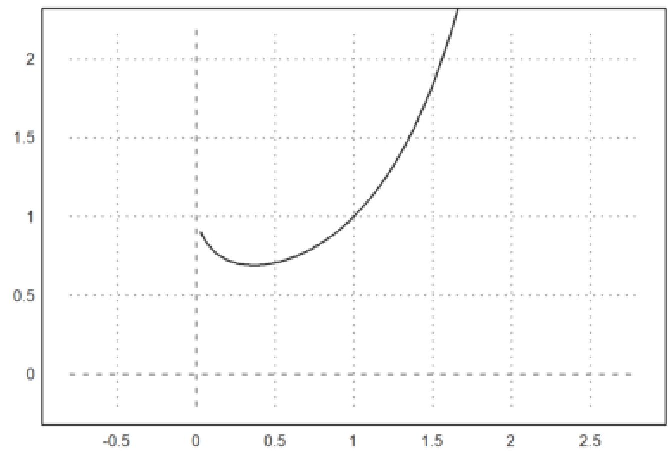
Anda cukup menandai gambar apa saja dan menyalinnya ke clipboard dengan Ctrl-C. Tentu saja, Anda juga dapat mengekspor grafik saat ini dengan fungsi di menu File.

Fungsi atau ekspresi di plot2d dievaluasi secara adaptif. Agar lebih cepat, nonaktifkan plot adaptif dengan <adaptive dan tentukan jumlah subinterval dengan n=... Hal ini hanya diperlukan dalam kasus yang jarang terjadi.

```
>plot2d("sign(x)*exp(-x^2)",-1,1,<adaptive,n=10000):
```

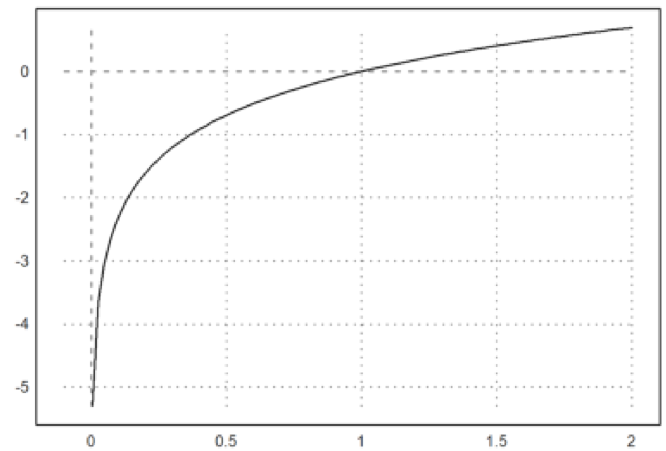



```
>plot2d("x^x",r=1.2,cx=1,cy=1):
```



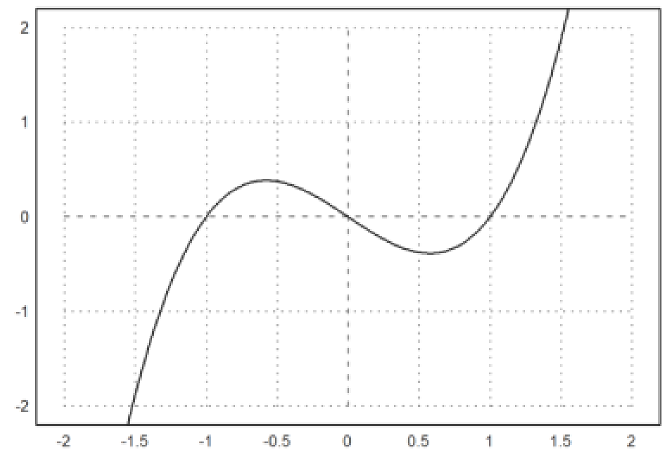
Perhatikan bahwa x^x tidak ditentukan untuk x<=0. Fungsi plot2d menangkap kesalahan ini, dan mulai membuat plot segera setelah fungsinya ditentukan. Ini berfungsi untuk semua fungsi yang mengembalikan NAN di luar jangkauan definisinya.

```
>plot2d("log(x)",-0.1,2):
```

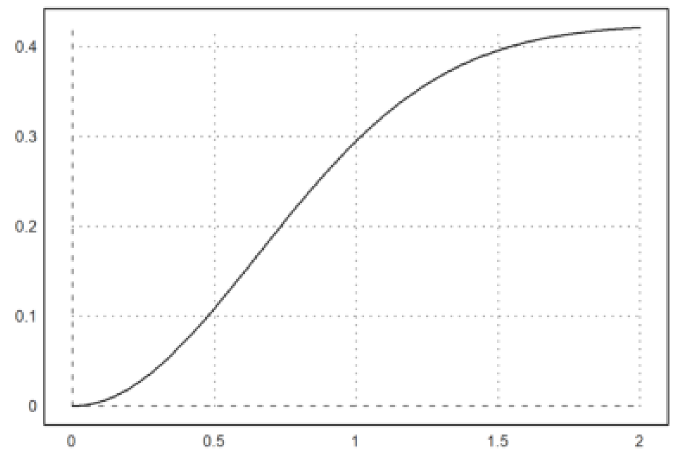


Parameter square=true (atau >square) memilih rentang y secara otomatis sehingga hasilnya adalah jendela plot persegi. Perhatikan bahwa secara default, Euler menggunakan spasi persegi di dalam jendela plot.

```
>plot2d("x^3-x",>square):
```

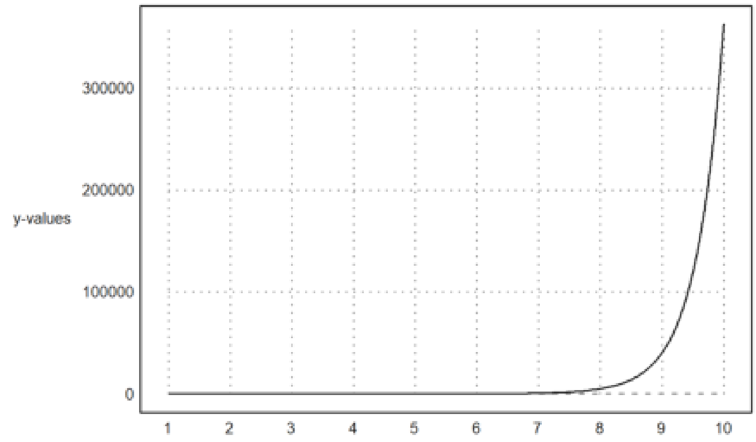



```
>plot2d(''integrate("sin(x)*exp(-x^2)",0,x)'',0,2): // plot integral
```



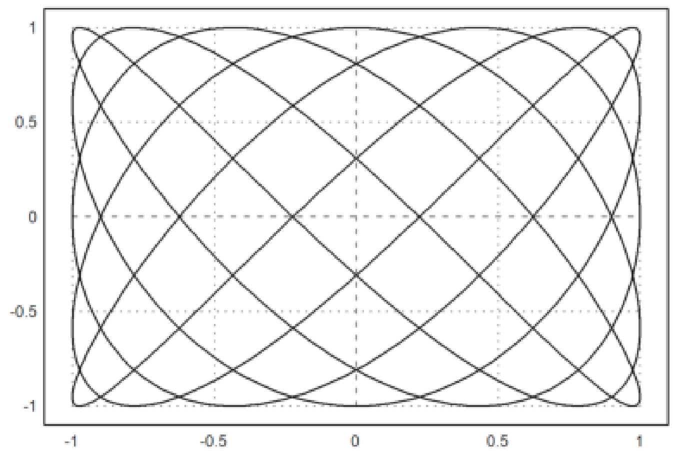
Jika Anda memerlukan lebih banyak ruang untuk label y, panggil shrinkwindow() dengan parameter lebih kecil, atau tetapkan nilai positif untuk "lebih kecil" di plot2d().

```
>plot2d("gamma(x)",1,10,yl="y-values",smaller=6,<vertical):
```

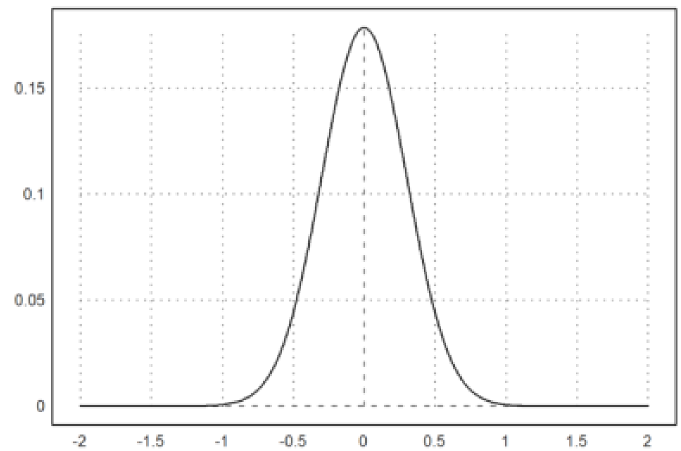


Ekspresi simbolik juga dapat digunakan karena disimpan sebagai ekspresi string sederhana.

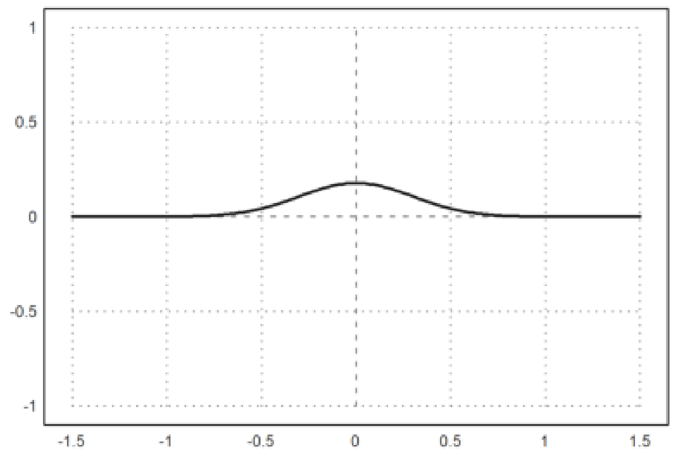
```
>x=linspace(0,2pi,1000); plot2d(sin(5x),cos(7x)):
```



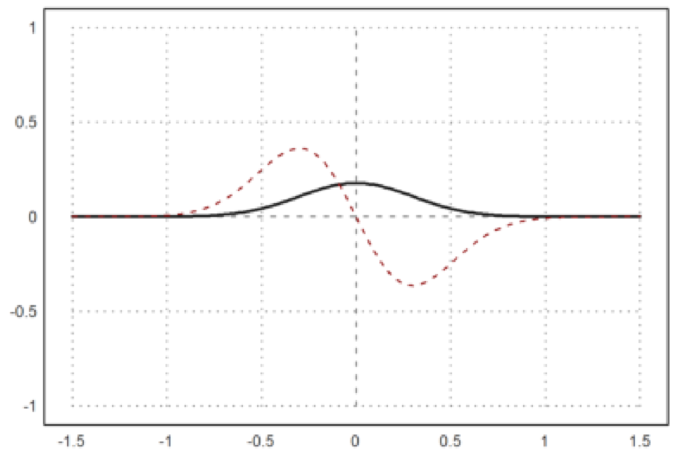

```
>a:=5.6; expr &= exp(-a*x^2)/a; // define expression
>plot2d(expr,-2,2): // plot from -2 to 2
```



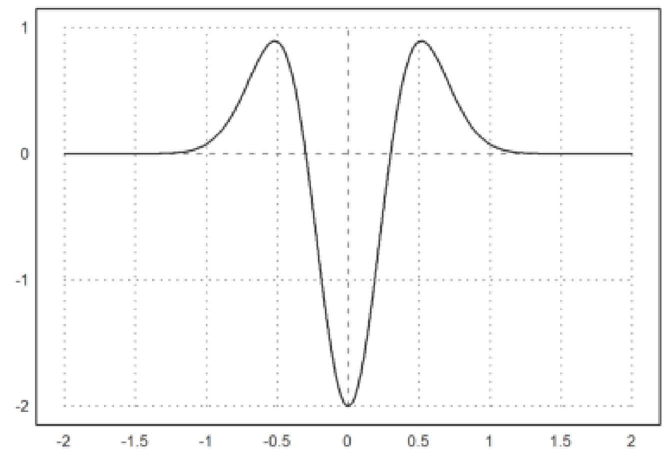

```
>plot2d(expr,r=1,thickness=2): // plot in a square around (0,0)
```



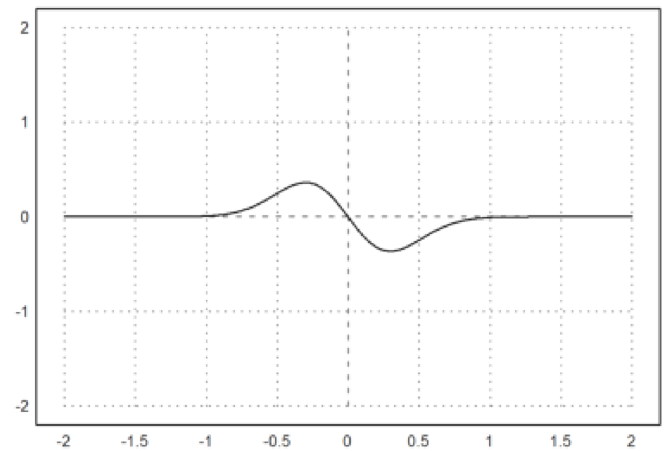

```
>plot2d(&diff(expr,x),>add,style="--",color=red): // add another plot
```



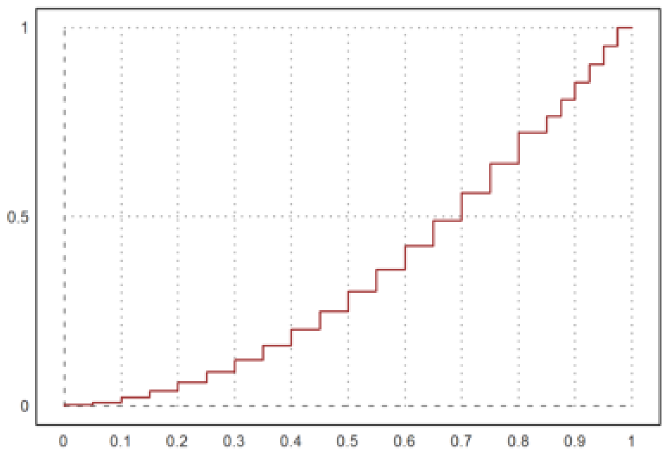

```
>plot2d(&diff(expr,x,2),a=-2,b=2,c=-2,d=1): // plot in rectangle
```



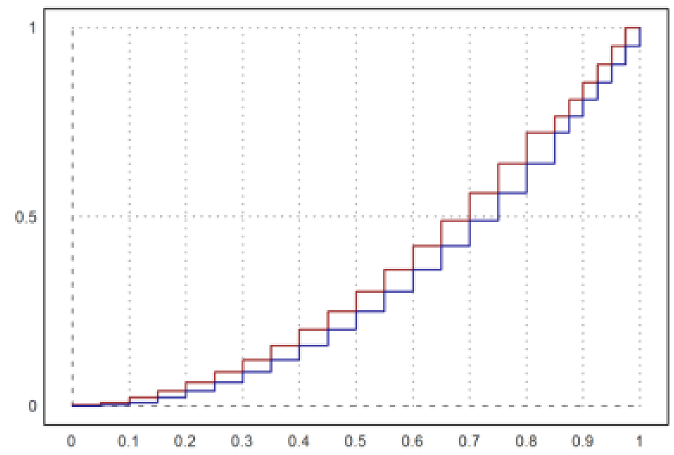

```
>plot2d(&diff(expr,x),a=-2,b=2,>square): // keep plot square
```




```
>plot2d("x^2",0,1,steps=1,color=red,n=10):
```




```
>plot2d("x^2",>add,steps=2,color=blue,n=10):
```



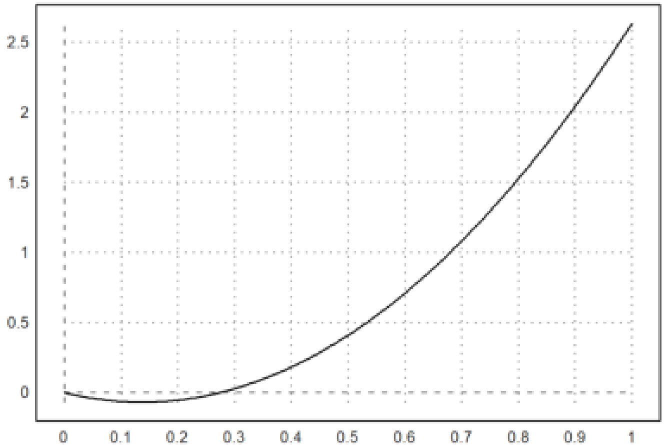

```
>a=0.3; plot2d("f",0,1;a): // plot with a=0.3
```

```
Function f needs at least 4 arguments!
Use: f (K, R, P, n)
 %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());
```

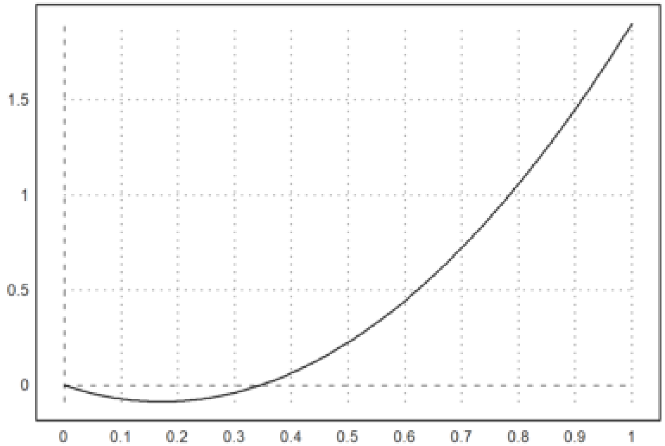
Fungsi plot yang paling penting untuk plot planar adalah plot2d(). Fungsi ini diimplementasikan dalam bahasa Euler di file "plot.e", yang dimuat di awal program.

Berikut beberapa contoh penggunaan suatu fungsi. Seperti biasa di EMT, fungsi yang berfungsi untuk fungsi atau ekspresi lain, Anda bisa meneruskan parameter tambahan (selain x) yang bukan variabel global ke fungsi dengan parameter titik koma atau dengan kumpulan panggilan.

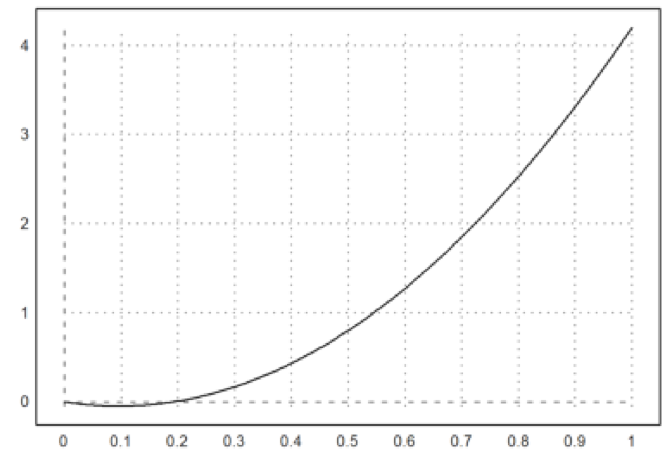
```
>function f(x,a) := (x^2/a+a*x^2-x); // define a function
>a=0.3; plot2d("f",0,1;a): // plot with a=0.3
```



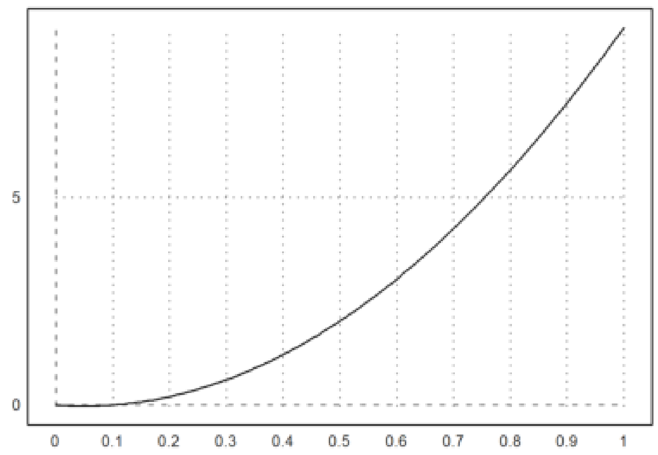
```
>plot2d("f",0,1;0.4): // plot with a=0.4
```



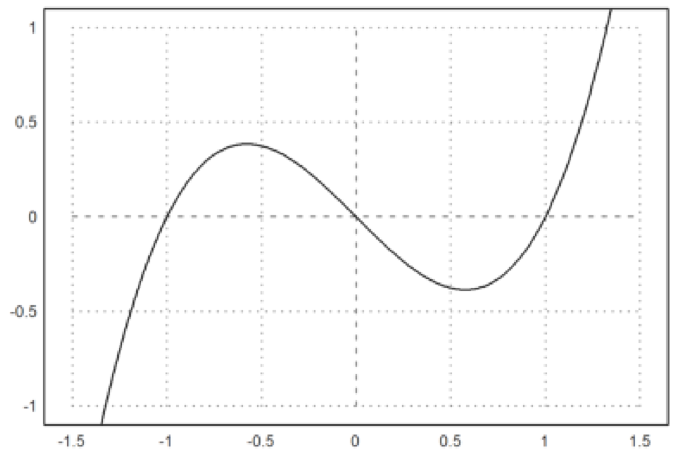

```
>plot2d({{"f",0.2}},0,1): // plot with a=0.2
```




```
>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 ringkasan fungsi yang diterima

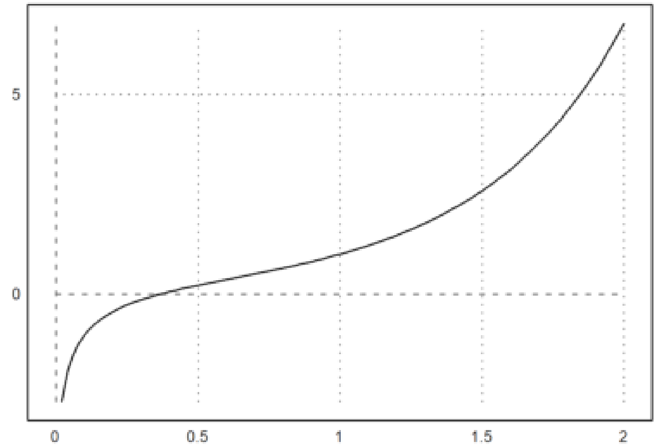
- ekspresi atau ekspresi simbolik di x
- fungsi atau fungsi simbolik dengan nama "f"
- fungsi simbolik hanya dengan nama f

Fungsi plot2d() juga menerima fungsi simbolik. Untuk fungsi simbolik, namanya saja yang berfungsi.

```
>function f(x) &= diff(x^x,x)
```

```
 x
x  (log(x) + 1)
```

```
>plot2d(f,0,2):
```

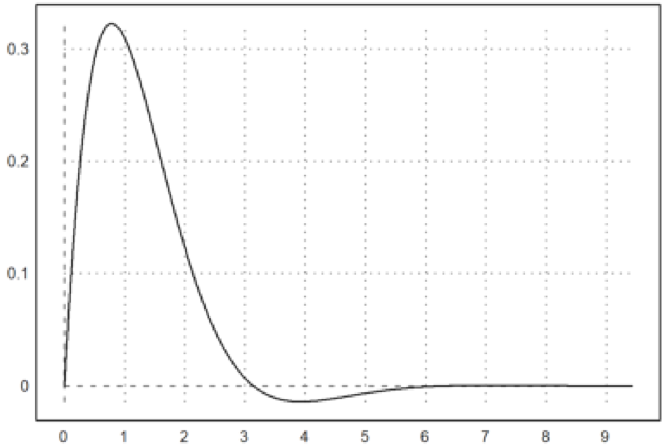


Tentu saja, untuk ekspresi atau ekspresi simbolik, nama variabel sudah cukup untuk memplotnya.

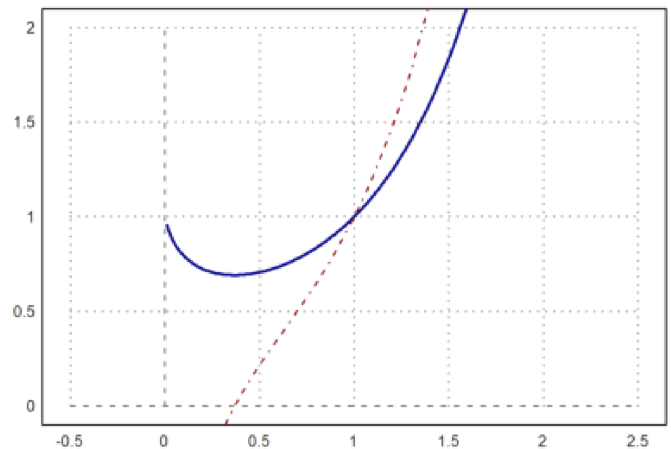
```
>expr &= sin(x)*exp(-x)
```

```
                                    - 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="-.-"):
```



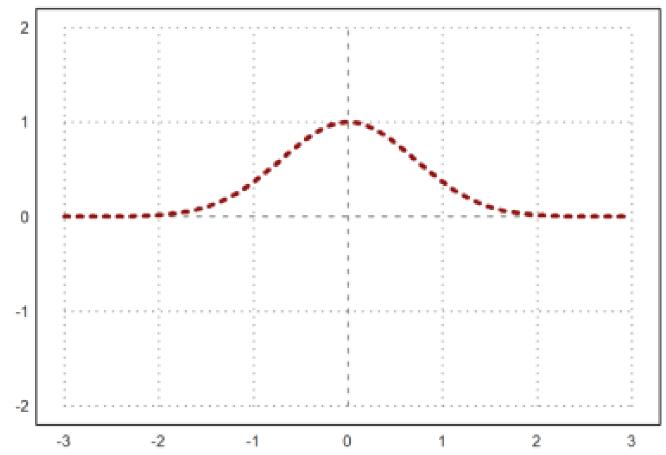
Untuk gaya garis ada berbagai pilihan.

- gaya=”...”. Pilih dari ”-”, ”–”, ”-.”, ”.”, ”.-.”, ”-.-”.
- Warna: Lihat di bawah untuk warna.
- ketebalan: Defaultnya adalah 1.

Warna dapat dipilih sebagai salah satu warna default, atau sebagai warna RGB.

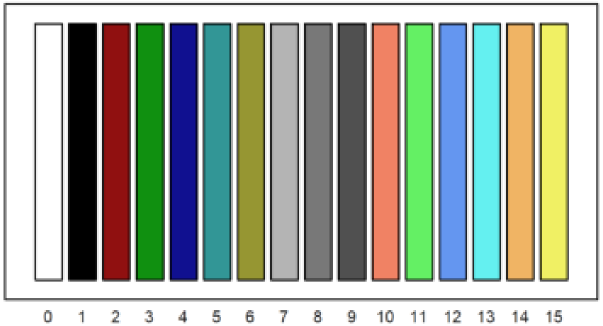
- 0..15: indeks warna default.
- konstanta warna: putih, hitam, merah, hijau, biru, cyan, zaitun, abu-abu muda, abu-abu, abu-abu tua, oranye, hijau muda, pirus, biru muda, oranye muda, kuning
- rgb(merah,hijau,biru): parameternya real di [0,1].

```
>plot2d("exp(-x^2)",r=2,color=red,thickness=3,style="--"):
```



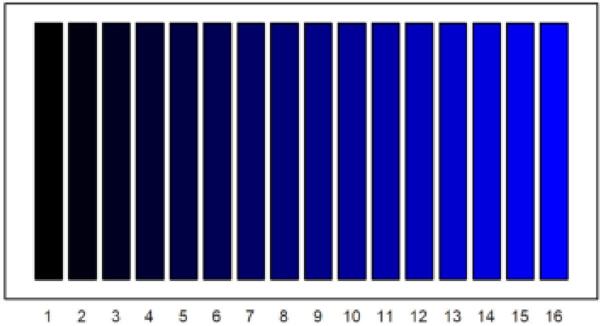
Berikut adalah tampilan warna EMT yang telah ditentukan sebelumnya.

```
>aspect(2); columnsplot(ones(1,16),lab=0:15,grid=0,color=0:15):
```



Tapi Anda bisa menggunakan warna apa saja.

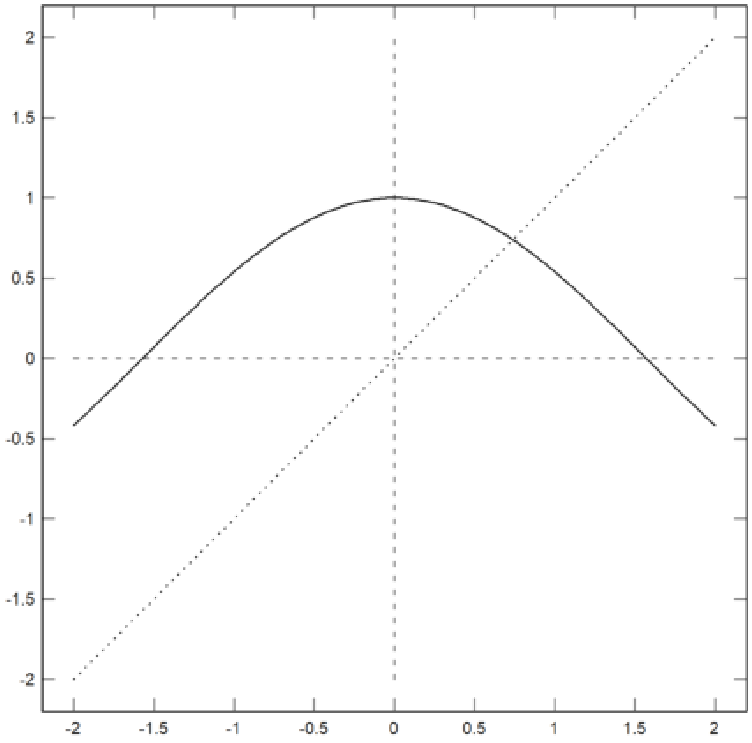
```
>columnsplot(ones(1,16),grid=0,color=rgb(0,0,linspace(0,1,15))):
```



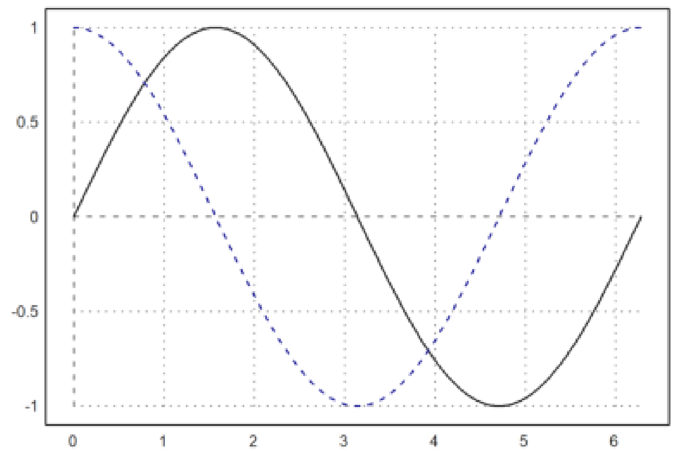
Menggambar Beberapa Kurva pada bidang koordinat yang sama

Plot lebih dari satu fungsi (multiple function) ke dalam satu jendela dapat dilakukan dengan berbagai cara. Salah satu metodenya adalah menggunakan >add untuk beberapa panggilan ke plot2d secara keseluruhan, kecuali panggilan pertama. Kami telah menggunakan fitur ini pada contoh di atas.

```
>aspect(); plot2d("cos(x)",r=2,grid=6); plot2d("x",style=".",>add):
```

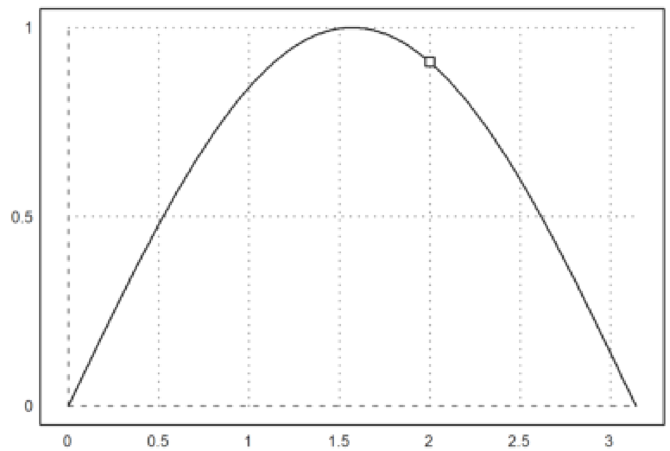



```
>aspect(1.5); plot2d("sin(x)",0,2pi); plot2d("cos(x)",color=blue,style="--",>add):
```



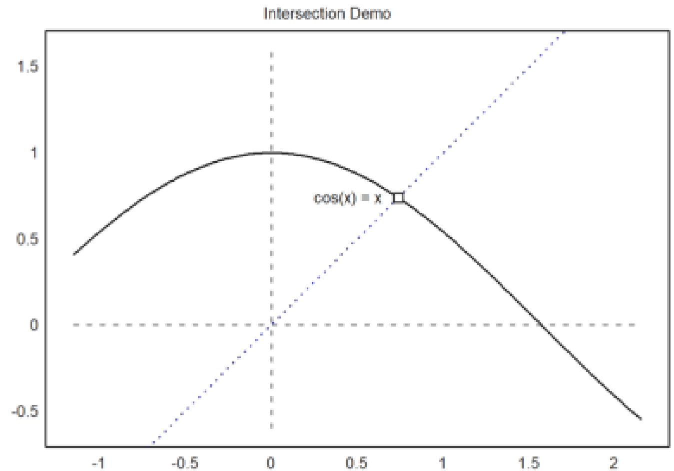
Salah satu kegunaan >add adalah untuk menambahkan titik pada kurva.

```
>plot2d("sin(x)",0,pi); plot2d(2,sin(2),>points,>add):
```



Kita tambahkan titik perpotongan dengan label (pada posisi ”cl” untuk kiri tengah), dan masukkan hasilnya ke dalam buku catatan. Kami juga menambahkan judul pada 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):
```



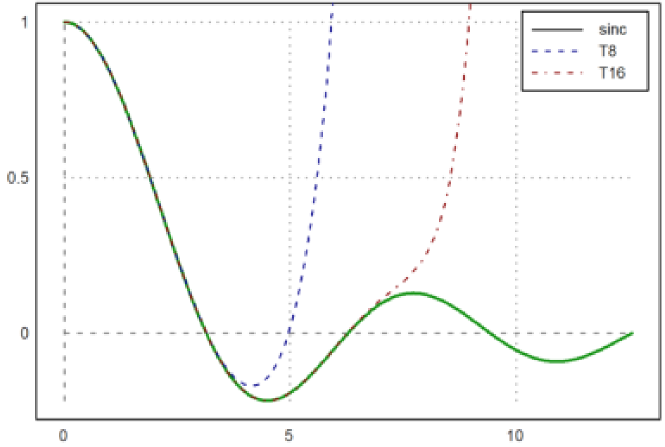
Dalam demo berikut, kita memplot fungsi sin(x)=sin(x)/x dan ekspansi Taylor ke-8 dan ke-16. Kami menghitung perluasan ini menggunakan Maxima melalui ekspresi simbolik.
Plot ini dilakukan dalam perintah multi-baris berikut dengan tiga panggilan ke plot2d(). Yang kedua dan ketiga memiliki kumpulan tanda >add, yang membuat plot menggunakan rentang sebelumnya.

Kami menambahkan kotak label yang menjelaskan fungsinya.

```
>$taylor(sin(x)/x,x,0,4)
```

$$\frac{x^4}{120} - \frac{x^2}{6} + 1$$

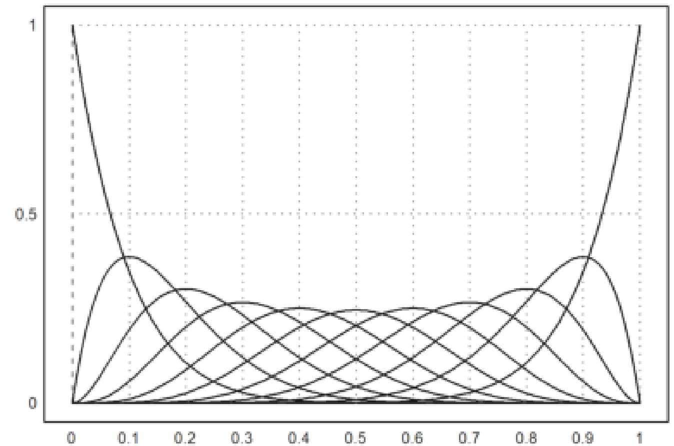
```
>plot2d("sinc(x)",0,4pi,color=green,thickness=2); ...
>  plot2d(&taylor(sin(x)/x,x,0,8),>add,color=blue,style="--"); ...
>  plot2d(&taylor(sin(x)/x,x,0,16),>add,color=red,style="-.-"); ...
>  labelbox(["sinc","T8","T16"],styles=["-","--","-.-"], ...
>    colors=[black,blue,red]):
```



Dalam contoh berikut, kami menghasilkan Polinomial Bernstein.

$$B_i(x) = \binom{n}{i} x^i (1-x)^{n-i}$$

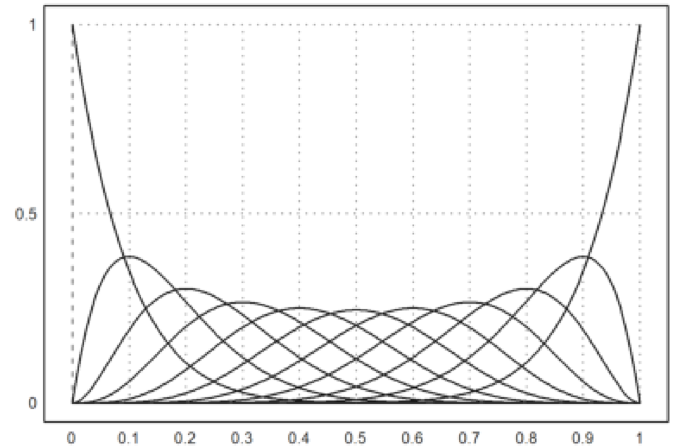
```
>plot2d("(1-x)^10",0,1); // plot first function
>for i=1 to 10; plot2d("bin(10,i)*x^i*(1-x)^(10-i)",>add); end;
>insimg;
```



Cara kedua adalah dengan menggunakan pasangan matriks bernilai x dan matriks bernilai y yang berukuran sama.

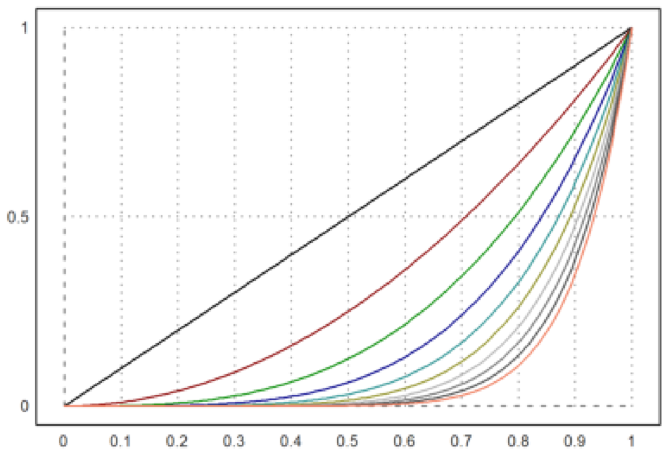
Kami menghasilkan matriks nilai dengan satu Polinomial Bernstein di setiap baris. Untuk ini, kita cukup menggunakan vektor kolom i. Lihat pendahuluan tentang bahasa matriks untuk mempelajari lebih detail.

```
>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):
```



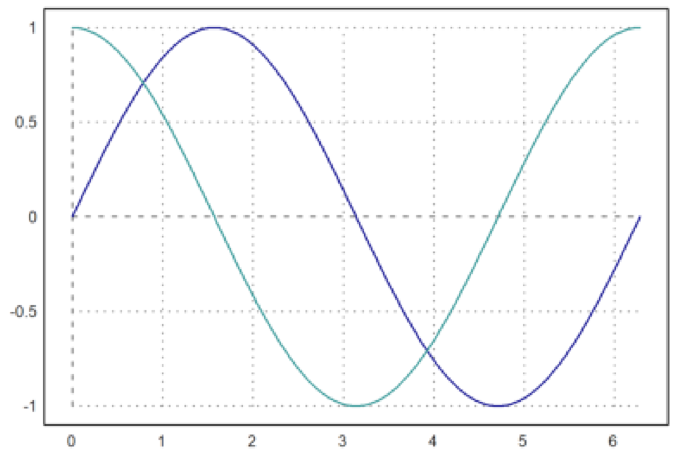
Perhatikan bahwa parameter warna dapat berupa vektor. Kemudian setiap warna digunakan untuk setiap baris matriks.

```
>x=linspace(0,1,200); y=x^(1:10)'; plot2d(x,y,color=1:10):
```

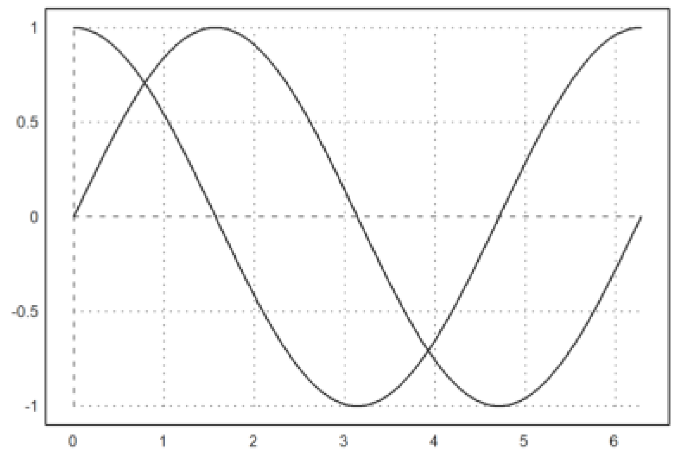


Metode lain adalah menggunakan vektor ekspresi (string). Anda kemudian dapat menggunakan susunan warna, susunan gaya, dan susunan ketebalan dengan panjang yang sama.

```
>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 menggunakan makelist() dan mxm2str().

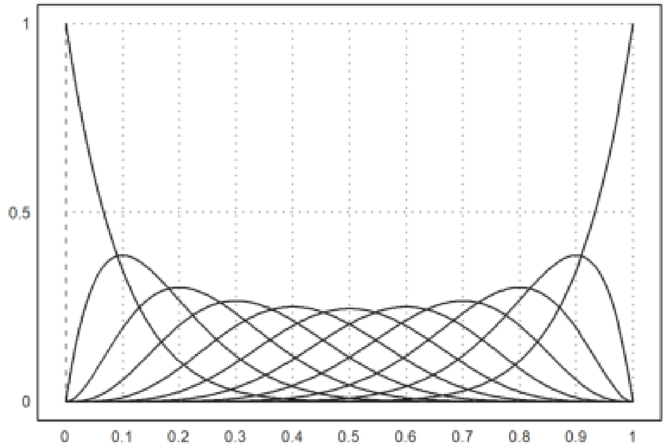
```
>v &= makelist(binomial(10,i)*x^i*(1-x)^(10-i),i,0,10) // make list
```

```
                 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
```

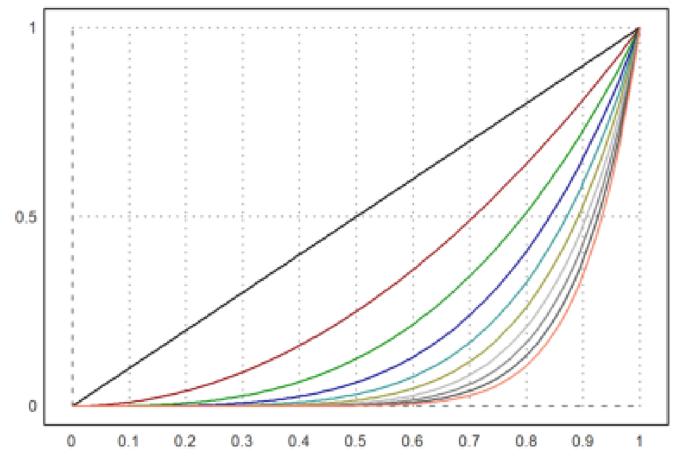


Alternatif lain adalah dengan menggunakan bahasa matriks Euler.

Jika suatu ekspresi menghasilkan matriks fungsi, dengan satu fungsi di setiap baris, semua fungsi tersebut akan diplot ke dalam satu plot.

Untuk ini, gunakan vektor parameter dalam bentuk vektor kolom. Jika array warna ditambahkan maka akan digunakan untuk setiap baris plot.

```
>n=(1:10)'; plot2d("x^n",0,1,color=1:10):
```

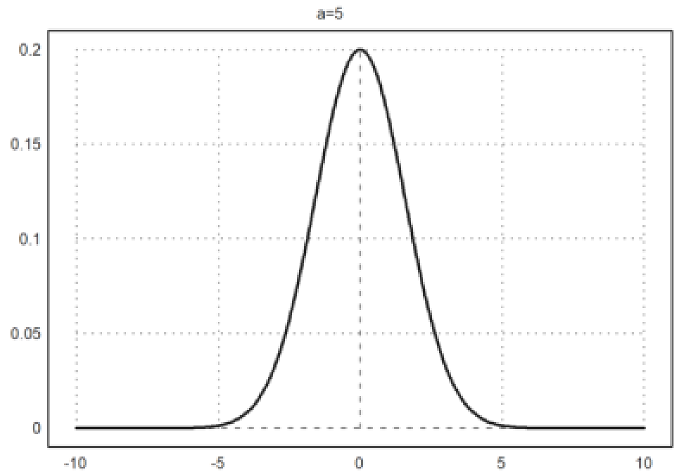


Ekspresi dan fungsi satu baris dapat melihat variabel global.

Jika Anda tidak dapat menggunakan variabel global, Anda perlu menggunakan fungsi dengan parameter tambahan, dan meneruskan parameter ini sebagai parameter titik koma.

Berhati-hatilah, untuk meletakkan semua parameter yang ditetapkan di akhir perintah plot2d. Dalam contoh ini kita meneruskan a=5 ke fungsi f, yang kita plot dari -10 hingga 10.

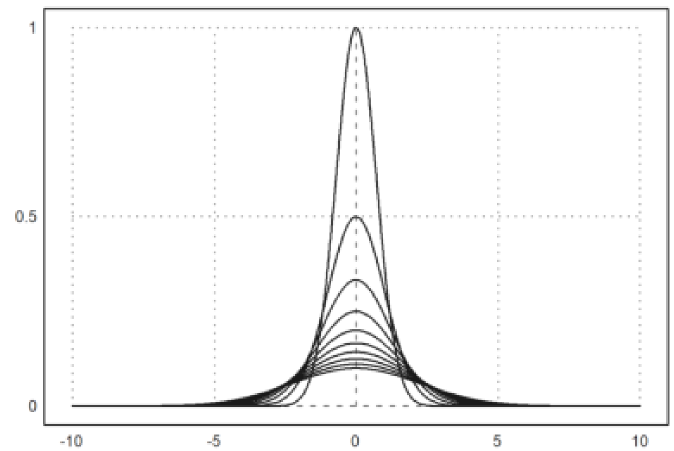
```
>function f(x,a) := 1/a*exp(-x^2/a); ...
>plot2d("f",-10,10;5,thickness=2,title="a=5"):
```



Alternatifnya, gunakan koleksi dengan nama fungsi dan semua parameter tambahan. Daftar khusus ini disebut kumpulan panggilan, dan ini adalah cara yang lebih disukai untuk meneruskan argumen ke suatu fungsi yang kemudian diteruskan sebagai argumen ke fungsi lain.

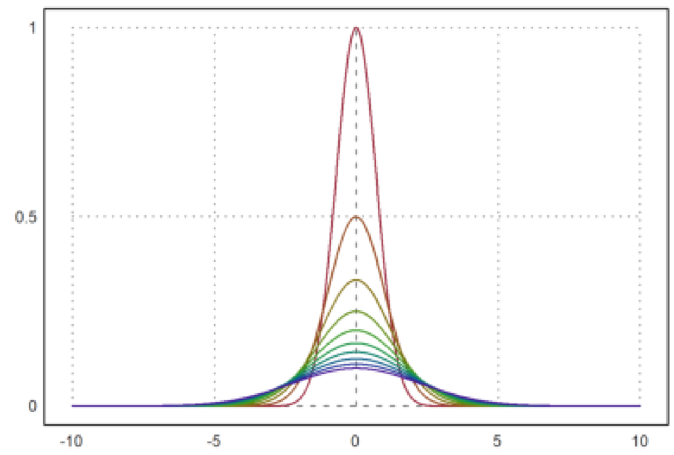
Pada contoh berikut, kita menggunakan loop untuk memplot beberapa fungsi (lihat tutorial tentang pemrograman loop).

```
>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 matriks f(x,a) merupakan 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)):
```



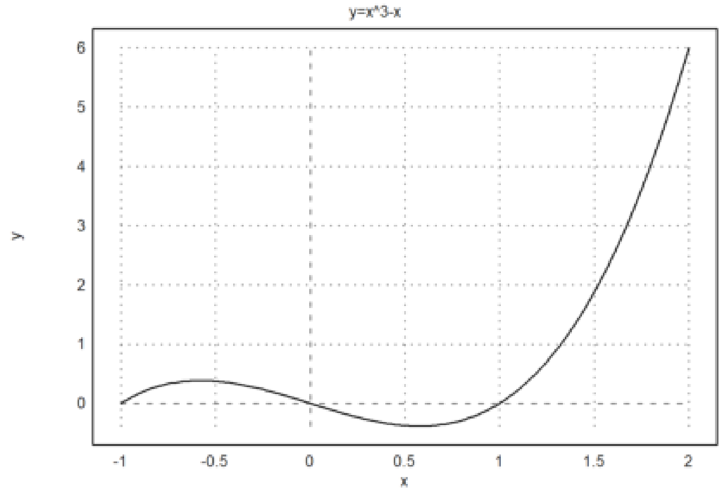
Label Teks

Dekorasi sederhana pun bisa

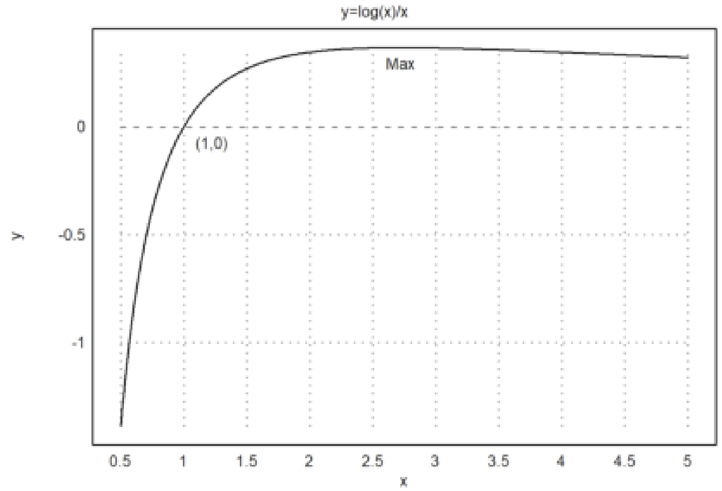
- judul dengan judul = "..."
- label x dan y dengan xl="...", yl="..."
- label teks lain dengan label("...",x,y)

Perintah label akan memplot ke plot saat ini pada koordinat plot (x,y). Hal ini memerlukan argumen posisional.

```
>plot2d("x^3-x",-1,2,title="y=x^3-x",yl="y",xl="x"):
```

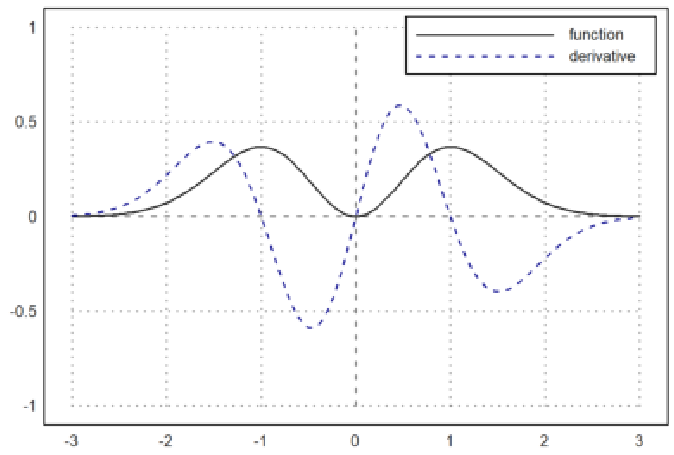



```
>expr := "log(x)/x"; ...
>  plot2d(expr,0.5,5,title="y="+expr,xl="x",yl="y"); ...
>  label("(1,0)",1,0); label("Max",E,expr(E),pos="lc"):
```



Ada juga fungsi labelbox(), yang dapat menampilkan fungsi dan teks. Dibutuhkan 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):
```

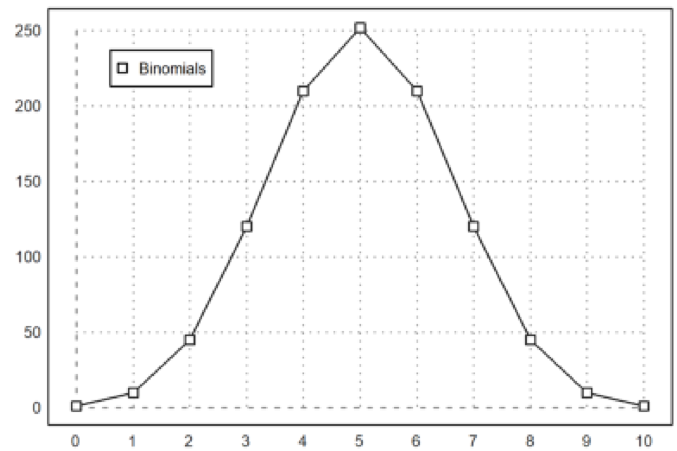


Kotak ini berlabuh di kanan atas secara default, tetapi >kiri berlabuh di kiri atas. Anda dapat memindahkannya ke tempat mana pun yang Anda suka. Posisi jangkar berada di pojok kanan atas kotak, dan angkanya merupakan pecahan dari ukuran jendela grafis. Lebarnya otomatis.

Untuk plot titik, kotak label juga berfungsi. Tambahkan parameter >points, atau vektor bendera, satu untuk setiap label.

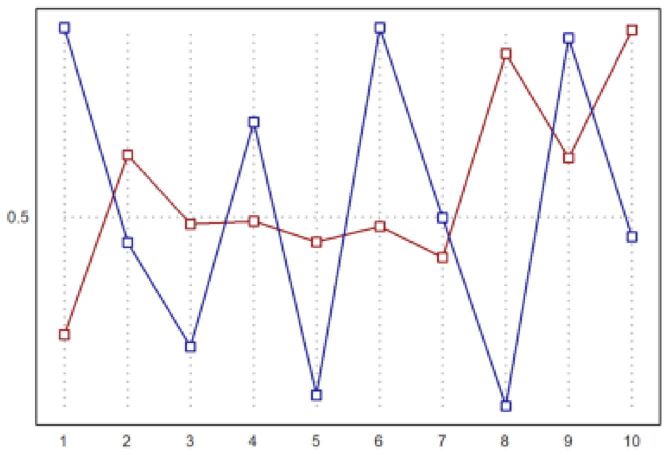
Pada contoh berikut, hanya ada satu fungsi. Jadi kita bisa menggunakan string sebagai pengganti vektor string. Kami mengatur warna teks menjadi hitam untuk contoh ini.

```
>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 di plot2d() warna dapat diatur untuk setiap baris plot. Masih banyak lagi plot khusus untuk keperluan statistik (lihat tutorial tentang statistik).

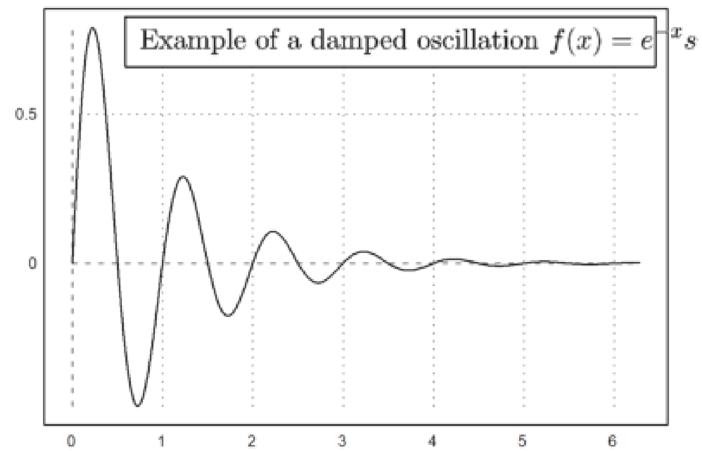
```
>statplot(1:10,random(2,10),color=[red,blue]):
```



Fitur serupa adalah fungsi textbox().

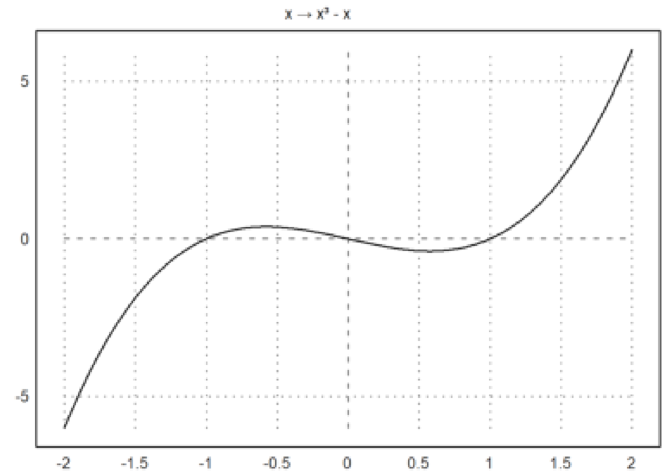
Lebarnya secara default adalah lebar maksimal baris teks. Tapi itu bisa diatur oleh pengguna juga.

```
>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 x)"),w=0.85):
```



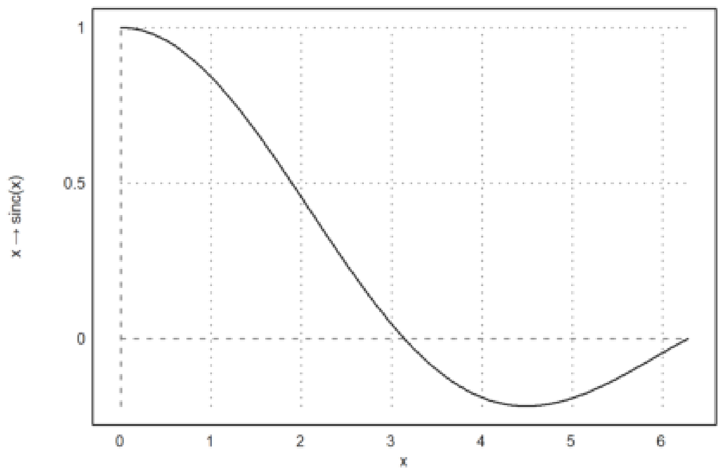
Label teks, judul, kotak label, dan teks lainnya dapat berisi string Unicode (lihat sintaks EMT untuk mengetahui lebih lanjut tentang string Unicode).

```
>plot2d("x^3-x",title=u"x &rarr; x&sup3; - x"):
```



Label pada sumbu x dan y bisa vertikal, begitu juga dengan sumbunya.

```
>plot2d("sinc(x)",0,2pi,xl="x",yl=u"x &rarr; sinc(x)",>vertical):
```



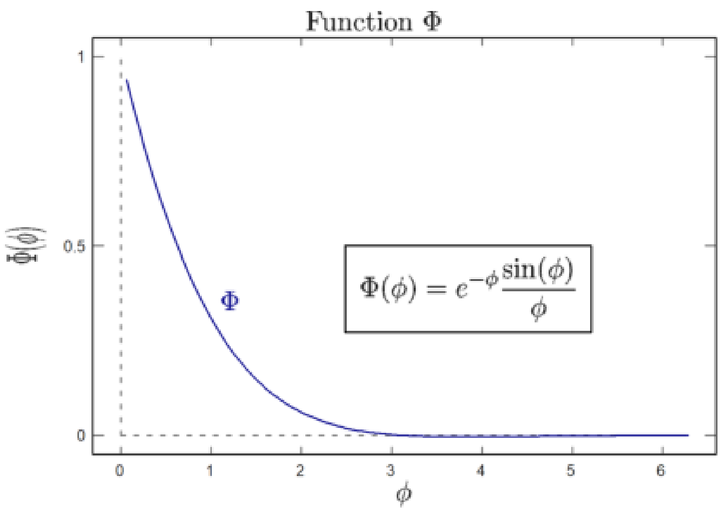
LaTeX

Anda juga dapat memplot rumus LaTeX jika Anda telah menginstal sistem LaTeX. Saya merekomendasikan MiKTeX. Jalur ke biner "lateks" dan "dvipng" harus berada di jalur sistem, atau Anda harus mengatur LaTeX di menu opsi.

Perhatikan, penguraian LaTeX lambat. Jika Anda ingin menggunakan LaTeX dalam plot animasi, Anda harus memanggil latex() sebelum loop satu kali dan menggunakan hasilnya (gambar dalam matriks RGB).

Pada plot berikut, kami menggunakan LaTeX untuk label x dan y, label, kotak label, dan judul plot.

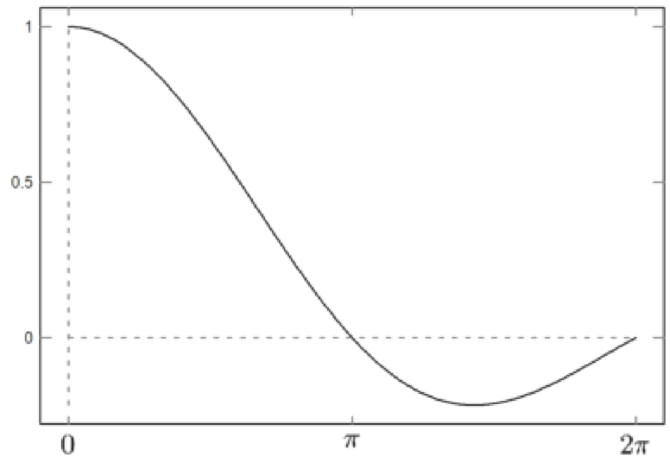
```
>plot2d("exp(-x)*sin(x)/x",a=0,b=2pi,c=0,d=1,grid=6,color=blue, ...
>  title=latex("\text{Function $\Phi$}"), ...
>  xl=latex("\phi"),yl=latex("\Phi(\phi)")); ...
>textbox( ...
>  latex("\Phi(\phi) = e^{-\phi} \frac{\sin(\phi)}{\phi}"),x=0.8,y=0.5); ...
>label(latex("\Phi",color=blue),1,0.4):
```



Seringkali, kita menginginkan spasi dan label teks yang tidak konformal pada sumbu x. Kita bisa menggunakan xaxis() dan yaxis() seperti yang akan kita tunjukkan nanti.

Cara termudah adalah membuat plot kosong dengan bingkai menggunakan grid=4, lalu menambahkan grid dengan ygrid() dan xgrid(). Pada contoh berikut, kami menggunakan tiga 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 fungsinya juga bisa digunakan.

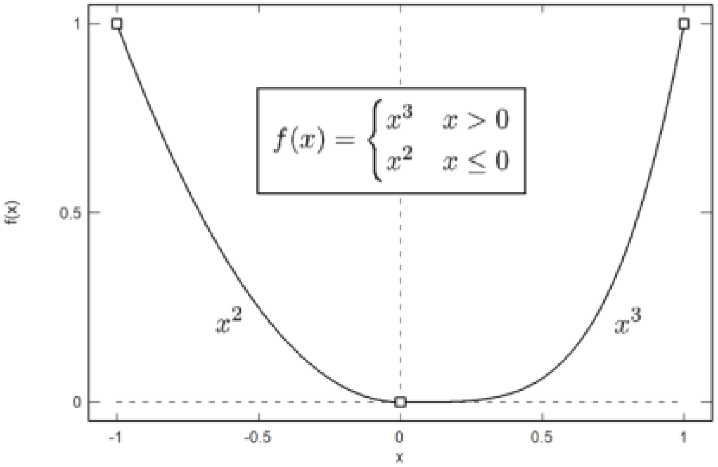
```
>function map f(x) ...
```

```
  if x>0 then return x^4
  else return x^2
  endif
  endfunction
```

Parameter ”peta” membantu menggunakan fungsi untuk vektor. Untuk plot, itu tidak perlu. Tapi untuk menunjukkan vektorisasi itu berguna, kita menambahkan beberapa poin penting ke plot di x=-1, x=0 dan x=1.

Pada plot berikut, kami juga memasukkan beberapa kode LaTeX. Kami menggunakannya untuk dua label dan kotak teks. Tentu saja, Anda hanya bisa menggunakannya 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):
```



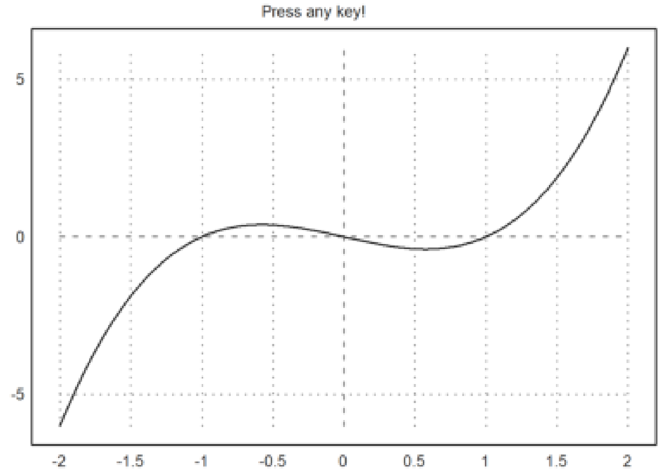
Saat memplot suatu fungsi atau ekspresi, parameter >pengguna memungkinkan pengguna untuk memperbesar dan menggeser plot dengan tombol kursor atau mouse. Pengguna bisa

- perbesar dengan + atau -
- pindahkan plot dengan tombol kursor
- pilih jendela plot dengan mouse
- atur ulang tampilan dengan spasi
- keluar dengan kembali

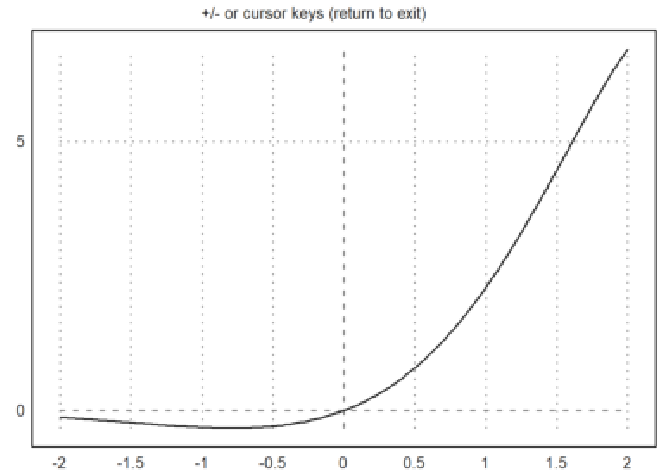
Tombol spasi akan mengatur ulang plot ke jendela plot aslinya.

Saat memplot data, flag >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 tentang pemrograman untuk detailnya).

Fungsi bawaan mousedrag() menunggu aktivitas mouse atau keyboard. Ini melaporkan mouse ke bawah, mouse digerakkan atau mouse ke atas, dan penekanan tombol. Fungsi dragpoints() memanfaatkan ini, dan memungkinkan pengguna menyeret titik mana pun dalam plot.

Kita membutuhkan fungsi plot terlebih dahulu. Misalnya, kita melakukan interpolasi pada 5 titik dengan polinomial. Fungsi tersebut harus diplot ke dalam area plot yang tetap.

```
>function plotf(xp,yp,select) ...
```

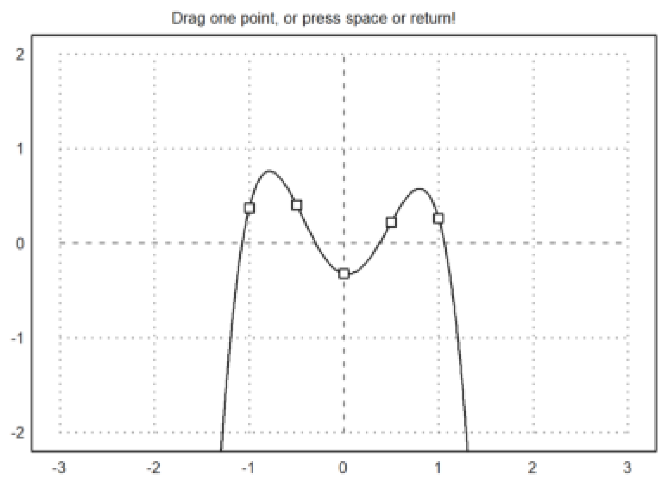
```
d=interp(xp,yp);
plot2d("interpval(xp,d,x)";d,xp,r=2);
plot2d(xp,yp,>points,>add);
if select>0 then
```

```
    plot2d(xp[select],yp[select],color=red,>points,>add);
  endif;
  title("Drag one point, or press space or return!");
endfunction
```

Perhatikan parameter titik koma di plot2d (d dan xp), yang diteruskan ke evaluasi fungsi interp(). Tanpa ini, kita harus menulis fungsi plotinterp() terlebih dahulu, mengakses nilainya secara global.

Sekarang kita menghasilkan beberapa nilai acak, dan membiarkan pengguna menyeret titiknya.

```
>t=-1:0.5:1; dragpoints("plotf",t,random(size(t))-0.5):
```



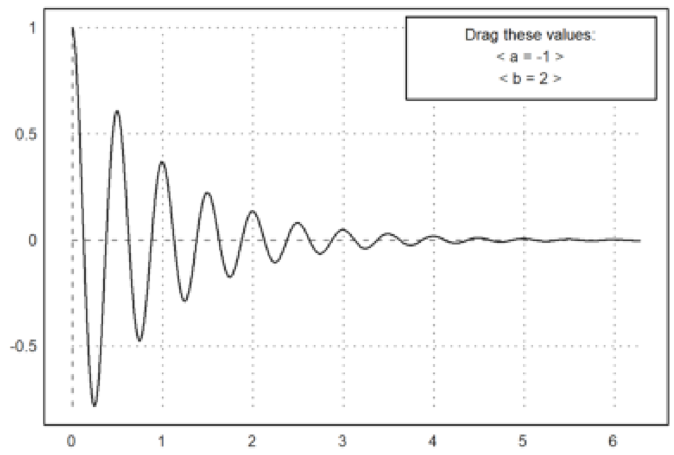
Ada juga fungsi yang memplot fungsi lain bergantung pada vektor parameter, dan memungkinkan pengguna menyesuaikan parameter ini.

Pertama kita membutuhkan fungsi plot.

```
>function plotf([a,b]) := plot2d("exp(a*x)*cos(2pi*b*x)",0,2pi;a,b);
```

Kemudian kita memerlukan nama untuk parameter, nilai awal dan matriks rentang nx2, opsional garis judul.
Ada penggeser interaktif, yang dapat menetapkan nilai oleh pengguna. Fungsi dragvalues() menyediakan ini.

```
>dragvalues("plotf",["a","b"],[-1,2],[[-2,2];[1,10]], ...
>  heading="Drag these values:",hcolor=black):
```



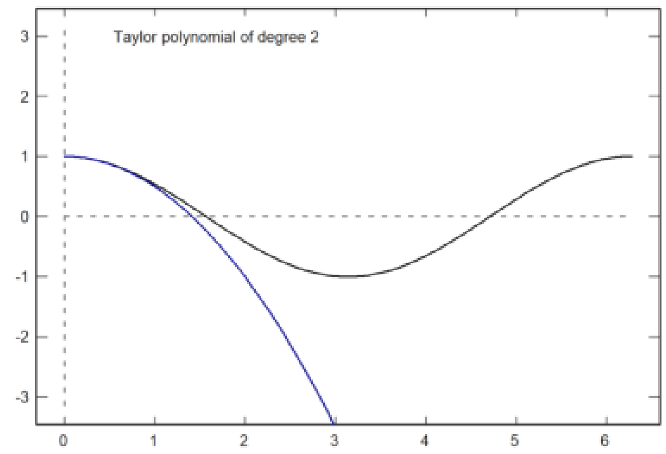
Dimungkinkan untuk membatasi nilai yang diseret menjadi bilangan bulat. Sebagai contoh, kita menulis fungsi plot, yang memplot polinomial Taylor berderajat n ke fungsi kosinus.

```
>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 izinkan derajat n bervariasi dari 0 hingga 20 dalam 20 perhentian. Hasil dragvalues() digunakan untuk memplot sketsa dengan n ini, dan untuk memasukkan 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 demonstrasi sederhana dari fungsinya. Pengguna dapat menggambar 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
```

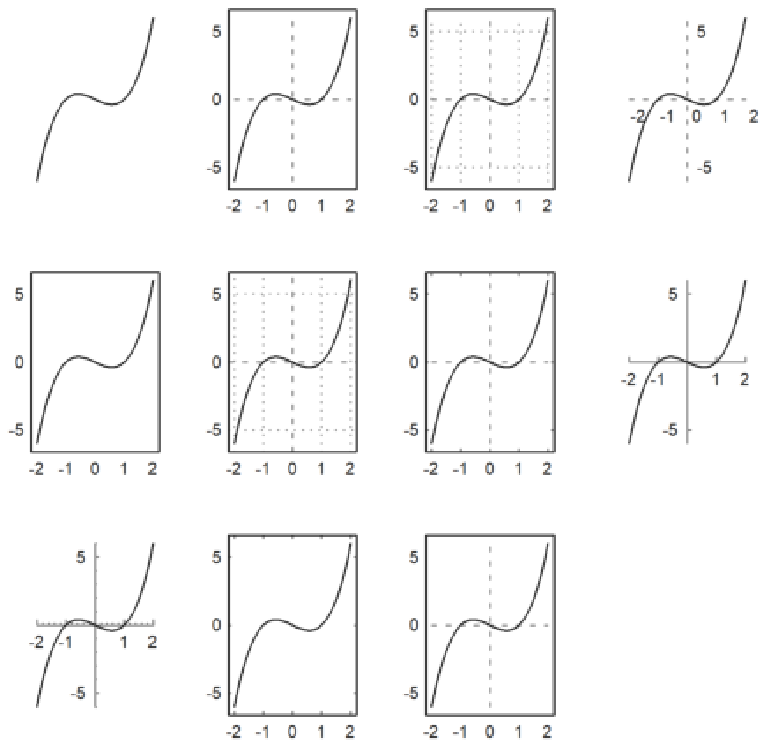
```
>dragtest // lihat hasilnya dan cobalah lakukan!
```

Gaya Plot 2D

Secara default, EMT menghitung tick sumbu otomatis dan menambahkan label ke setiap tick. Ini dapat diubah dengan parameter grid. Gaya default sumbu dan label dapat diubah. Selain itu, label dan judul dapat ditambahkan secara manual. Untuk menyetel 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
```

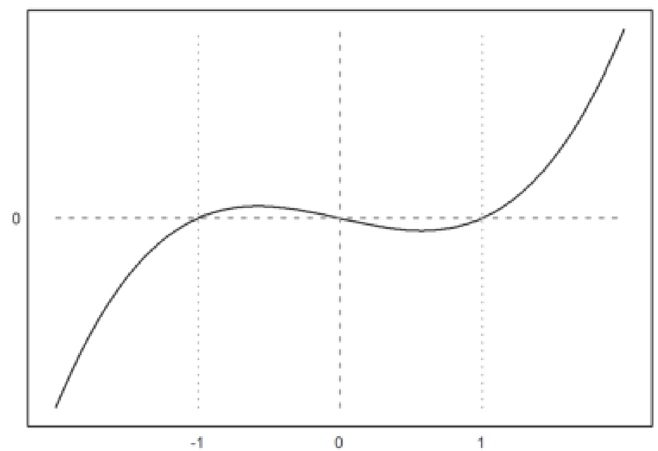
```
> figure(6); plot2d("x^3-x",grid=5); ... // default, but no margin
> figure(7); plot2d("x^3-x",grid=6); ... // axes only
> figure(8); plot2d("x^3-x",grid=7); ... // axes only, ticks at axis
> figure(9); plot2d("x^3-x",grid=8); ... // axes only, finer ticks at axis
> figure(10); plot2d("x^3-x",grid=9); ... // default, small ticks inside
> figure(11); plot2d("x^3-x",grid=10); ...// no ticks, axes only
> figure(0):
```



Parameter <frame mematikan frame, dan framecolor=blue mengatur frame menjadi warna biru.

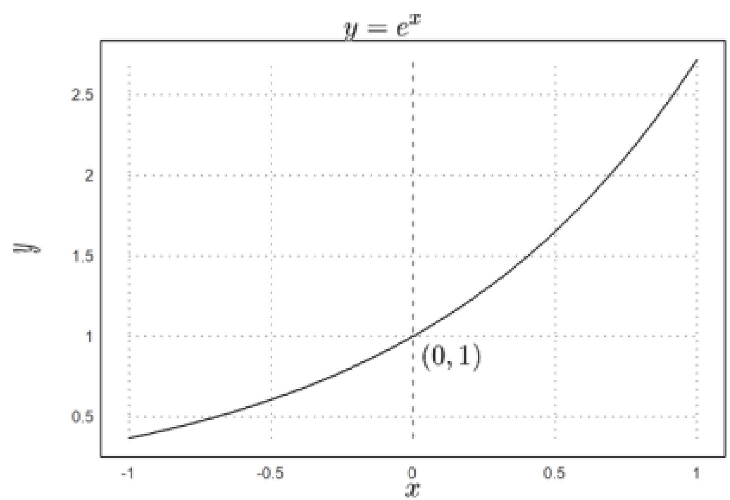
Jika Anda menginginkan tanda centang Anda sendiri, Anda dapat menggunakan style=0, dan menambahkan semuanya nanti.

```
>aspect(1.5);
>plot2d("x^3-x",grid=0); // plot
>frame; xgrid([-1,0,1]); ygrid(0): // add frame and grid
```



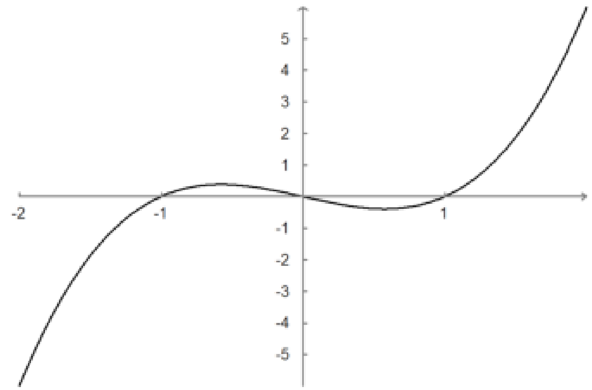
Untuk judul plot dan label sumbu, lihat contoh berikut.

```
>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 xaxis() dan yaxis().

```
>plot2d("x^3-x",<grid,<frame);
>xaxis(0,xx=-2:1,style="->"); yaxis(0,yy=-5:5,style="->"):
```

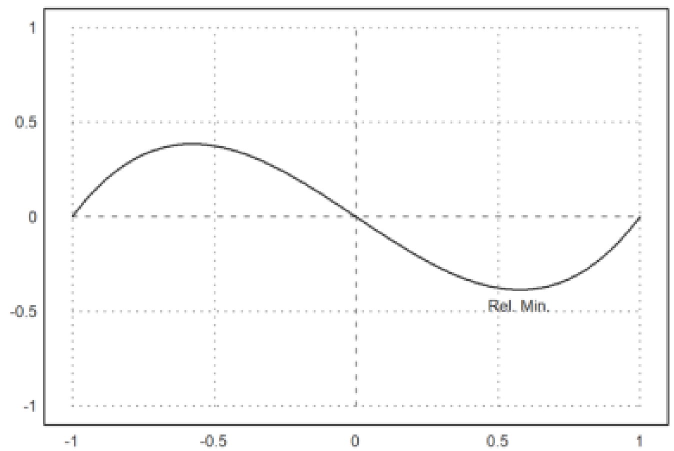


Teks pada plot dapat diatur dengan label(). Dalam contoh berikut, ”lc” berarti bagian tengah bawah. Ini menetapkan posisi label relatif terhadap koordinat plot.

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

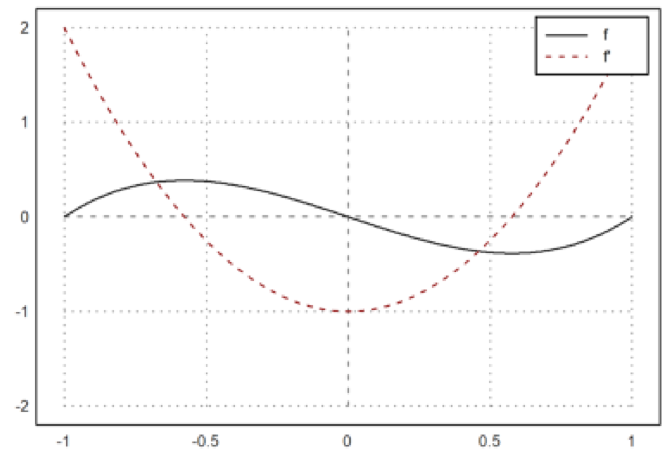
```
                                 3
                                x  - x
```

```
>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
```

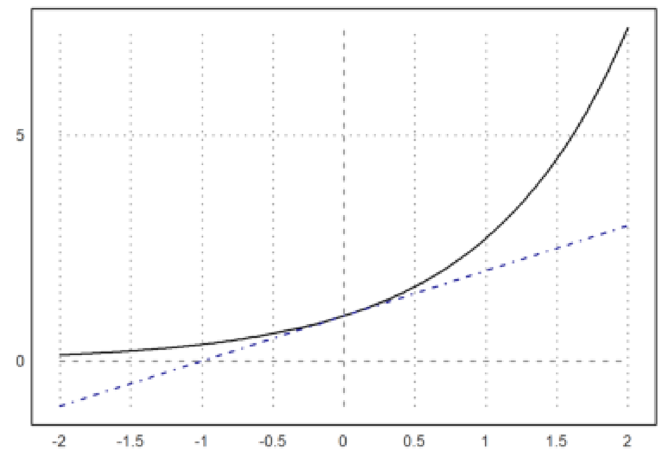


Ada juga kotak teks.

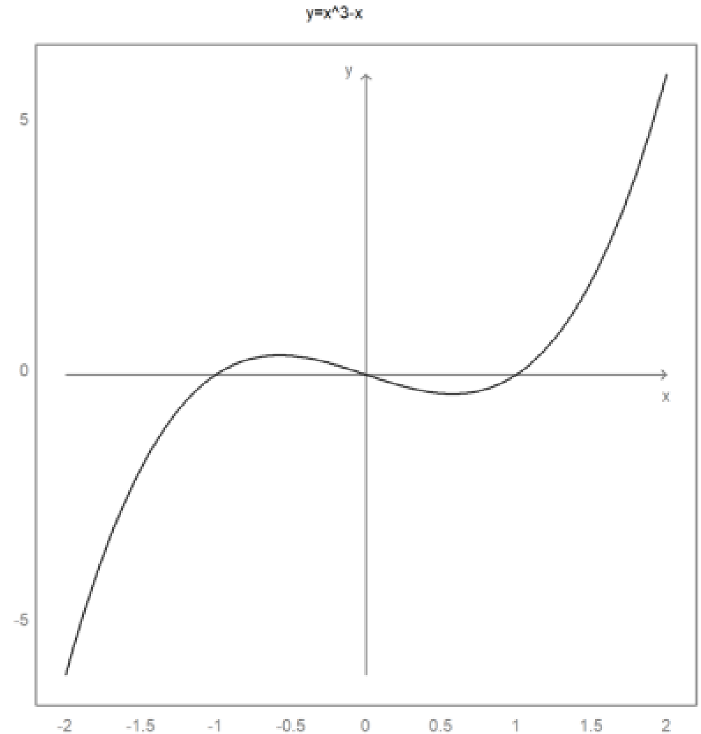
```
>plot2d(&f(x),-1,1,-2,2); // function
>plot2d(&diff(f(x),x),>add,style="--",color=red); // derivative
>labelbox(["f","f'"],["-","--"],[black,red]): // label box
```




```
>plot2d(["exp(x)","1+x"],color=[black,blue],style=["-","-.-"]):
```



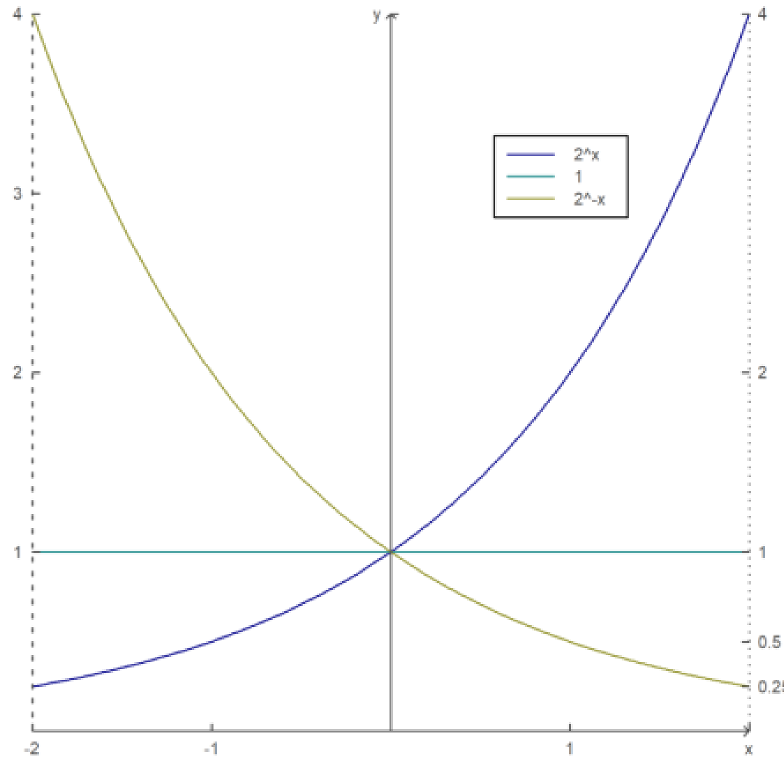

```
>gridstyle("->",color=gray,textcolor=gray,framecolor=gray);  ...
> plot2d("x^3-x",grid=1);   ...
> settitle("y=x^3-x",color=black); ...
> label("x",2,0,pos="bc",color=gray);  ...
> label("y",0,6,pos="cl",color=gray); ...
> reset():
```



Untuk kontrol lebih lanjut, sumbu x dan sumbu y dapat dilakukan secara manual.

Perintah fullwindow() memperluas jendela plot karena kita tidak lagi memerlukan tempat untuk label di luar jendela plot. Gunakan shrinkwindow() atau reset() untuk menyetel ulang ke default.

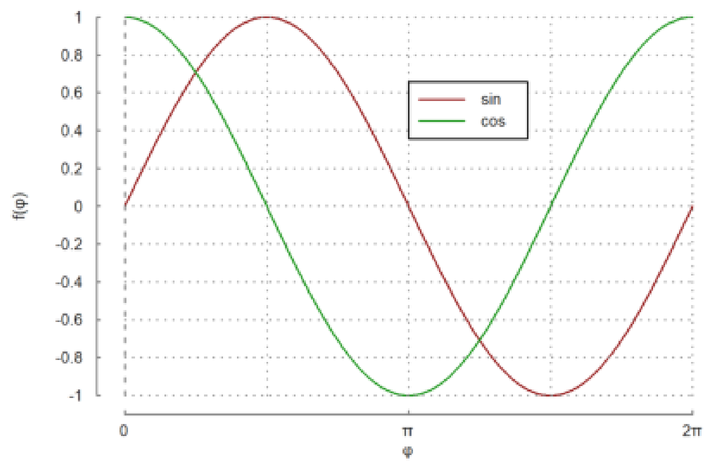
```
>fullwindow; ...
> gridstyle(color=darkgray,textcolor=darkgray); ...
> plot2d(["2^x","1","2^(-x)"],a=-2,b=2,c=0,d=4,<grid,color=4:6,<frame); ...
> xaxis(0,-2:1,style="->"); xaxis(0,2,"x",<axis); ...
> yaxis(0,4,"y",style="->"); ...
> yaxis(-2,1:4,>left); ...
> yaxis(2,2^(-2:2),style=".",<left); ...
> labelbox(["2^x","1","2^-x"],colors=4:6,x=0.8,y=0.2); ...
> reset:
```



Berikut adalah contoh lain, di mana string Unicode digunakan dan sumbunya berada 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;)"):
```



Jika x dan y adalah vektor data, maka data tersebut akan digunakan sebagai koordinat x dan y pada suatu kurva. Dalam hal ini, a, b, c, dan d, atau radius r dapat ditentukan, atau jendela plot akan menyesuaikan secara otomatis dengan data. Alternatifnya, >persegi dapat diatur untuk mempertahankan rasio aspek persegi.

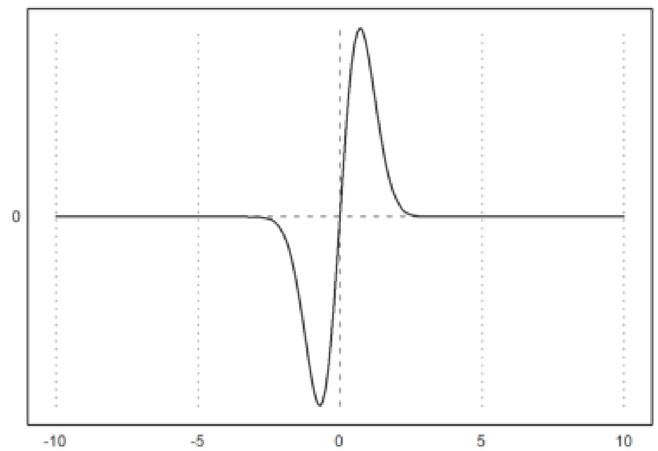
Merencanakan ekspresi hanyalah singkatan dari plot data. Untuk plot data, Anda memerlukan satu atau beberapa baris nilai x, dan satu atau beberapa baris nilai y. Dari rentang dan nilai x, fungsi plot2d akan menghitung data yang akan diplot, secara default dengan evaluasi fungsi yang adaptif. Untuk plot titik gunakan ">titik", untuk garis dan titik campuran gunakan ">addpoints".

Tapi Anda bisa 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):
```

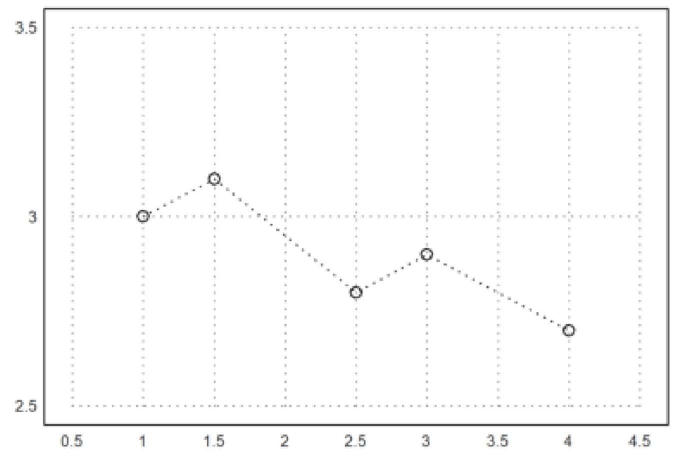


Data juga dapat diplot sebagai poin. Gunakan points=true untuk ini. Plotnya berfungsi seperti poligon, tetapi hanya menggambar sudutnya saja.

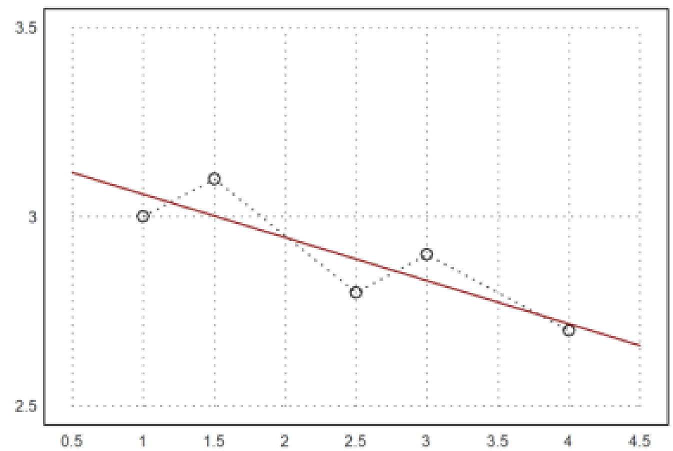
- style=”...”: Pilih dari ”[]”, ”<>”, ”o”, ”.”, ”..”, ”+”, ”*”, ”[]”, ”< >”, ”o”, ”..”, ””, ”|”.

Untuk memplot kumpulan titik, gunakan >titik. Jika warna merupakan vektor warna, masing-masing titik
mendapat warna berbeda. Untuk matriks koordinat dan vektor kolom, warna diterapkan pada 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
```



3.5
3
2.5
0.5
1
1.5
2
2.5
3
3.5
4
4.5

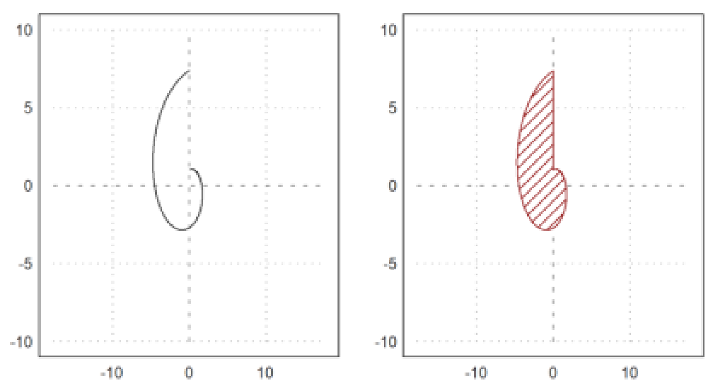
Menggambar Daerah Yang Dibatasi Kurva

Plot data sebenarnya berbentuk poligon. Kita juga dapat memplot kurva atau kurva terisi.

- terisi=benar mengisi plot.
- style=”...”: Pilih dari ””, ”/”, ”\”, ”\/”.
- Fillcolor : Lihat di atas untuk mengetahui warna yang tersedia.

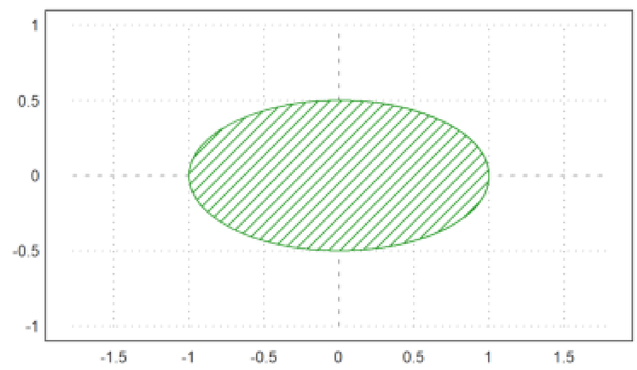
Warna isian ditentukan oleh argumen ”fillcolor”, dan pada <outline opsional, mencegah menggambar batas untuk semua gaya kecuali gaya default.

```
>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):
```

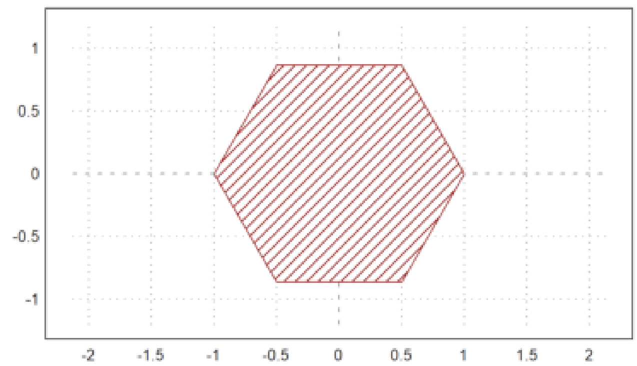


Dalam contoh berikut kita memplot elips terisi dan dua segi enam terisi menggunakan kurva tertutup dengan 6 titik dengan gaya isian berbeda.

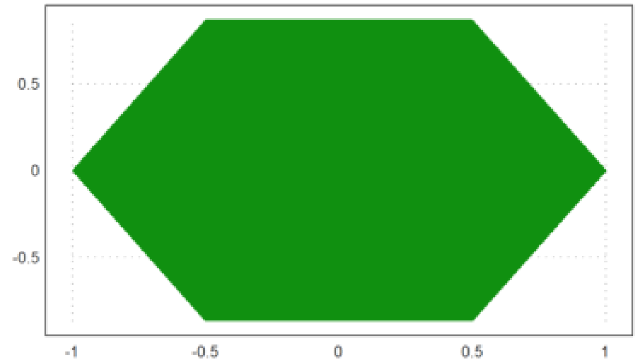
```
>x=linspace(0,2pi,1000); plot2d(sin(x),cos(x)*0.5,r=1,>filled,style="/"):
```




```
>t=linspace(0,2pi,6); ...
>plot2d(cos(t),sin(t),>filled,style="/",fillcolor=red,r=1.2):
```

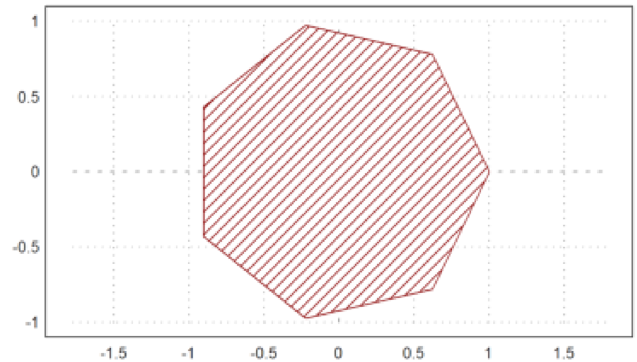



```
>t=linspace(0,2pi,6); plot2d(cos(t),sin(t),>filled,style="#"):
```



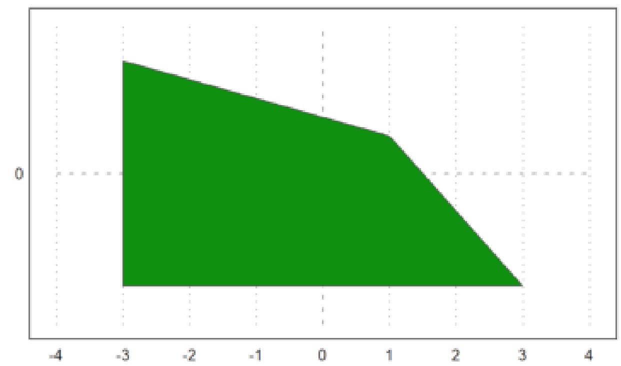
Contoh lainnya adalah septagon yang kita buat dengan 7 titik pada lingkaran satuan.

```
>t=linspace(0,2pi,7);  ...
> plot2d(cos(t),sin(t),r=1,>filled,style="/",fillcolor=red):
```



Berikut adalah himpunan nilai maksimal dari empat kondisi linier yang kurang dari atau sama dengan 3. Ini adalah A[k].v<=3 untuk semua baris A. Untuk mendapatkan 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):
```

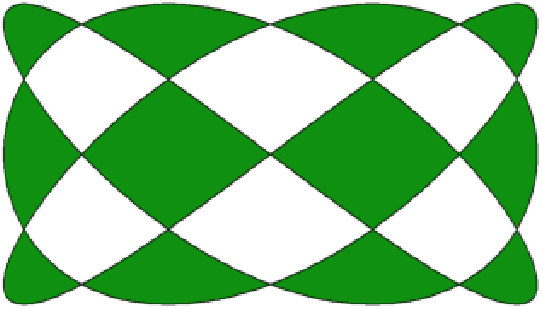


Poin utama dari bahasa matriks adalah memungkinkan pembuatan tabel fungsi dengan mudah.

```
>t=linspace(0,2pi,1000); x=cos(3*t); y=sin(4*t);
```

Kami sekarang memiliki nilai vektor x dan y. plot2d() dapat memplot nilai-nilai ini sebagai kurva yang menghubungkan titik-titik tersebut. Plotnya bisa diisi. Dalam hal ini ini menghasilkan hasil yang bagus karena aturan belitan, yang digunakan untuk isi.

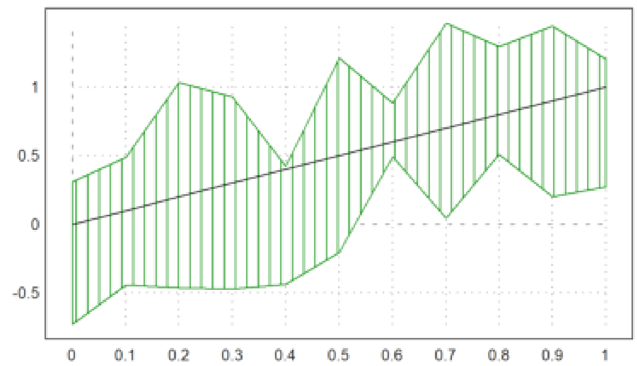
```
>plot2d(x,y,<grid,<frame,>filled):
```



Vektor interval diplot terhadap nilai x sebagai wilayah terisi
antara nilai interval yang lebih rendah dan lebih tinggi.

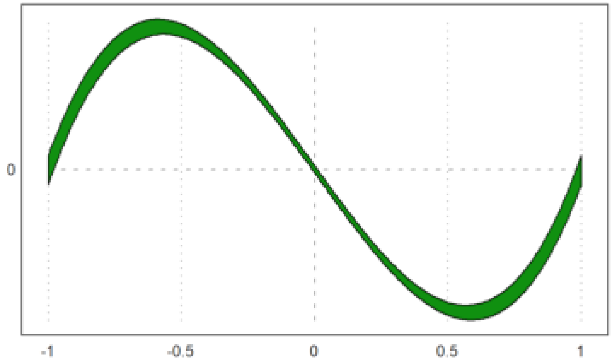
Hal ini dapat berguna untuk memplot kesalahan perhitungan. Tapi itu bisa
juga dapat digunakan untuk memplot kesalahan statistik.

```
>t=0:0.1:1; ...
> plot2d(t,interval(t-random(size(t)),t+random(size(t))),style="|");  ...
> plot2d(t,t,add=true):
```



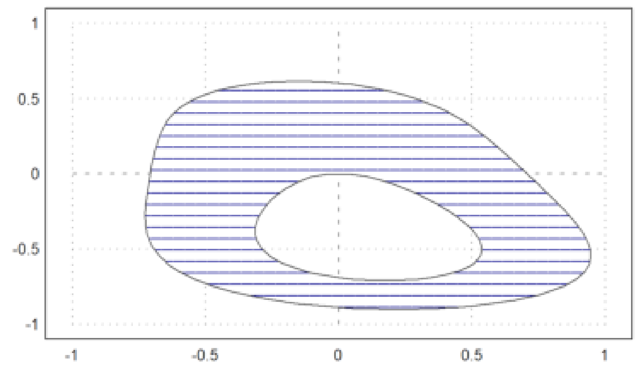
Jika x adalah vektor yang diurutkan, dan y adalah vektor interval, maka plot2d akan memplot rentang interval yang terisi pada bidang. Gaya isiannya sama dengan gaya poligon.

```
>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.

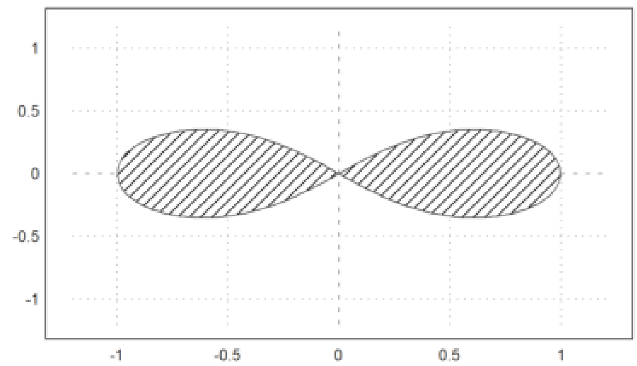
```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <= 1
```



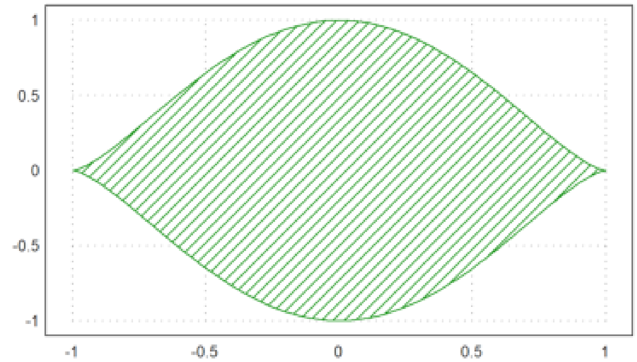
Kita juga dapat mengisi rentang nilai seperti

lateks: -1 \le (xˆ2+yˆ2)ˆ2-xˆ2+yˆ2 \le 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="/"):
```



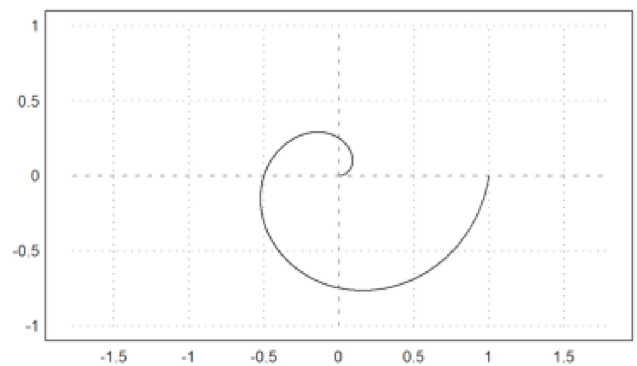
Nilai x tidak perlu diurutkan. (x,y) hanya menggambarkan sebuah kurva. Jika x diurutkan, kurva tersebut merupakan grafik suatu fungsi.

Dalam contoh berikut, kita memplot spiral

$$\gamma(t) = t \cdot (\cos(2\pi t), \sin(2\pi t))$$

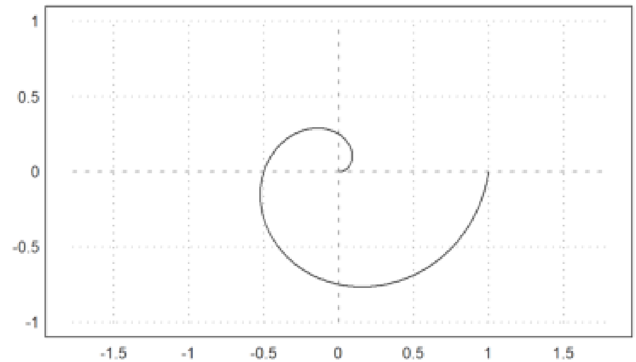
Kita perlu menggunakan banyak titik untuk tampilan yang halus atau fungsi adaptif() untuk mengevaluasi ekspresi (lihat fungsi adaptif() untuk lebih jelasnya).

```
>t=linspace(0,1,1000); ...
>plot2d(t*cos(2*pi*t),t*sin(2*pi*t),r=1):
```

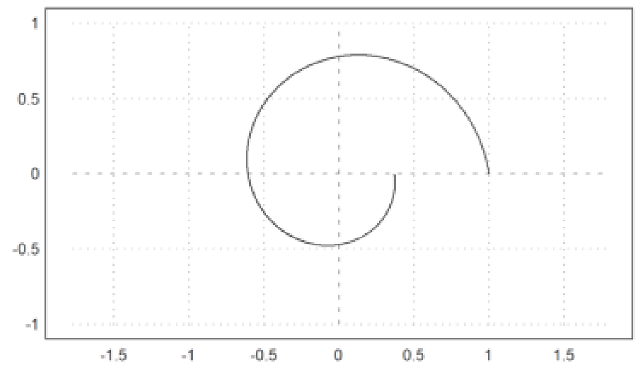


Sebagai alternatif, dimungkinkan untuk menggunakan dua ekspresi untuk kurva. Berikut ini plot kurva yang sama seperti di atas.

```
>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):
```



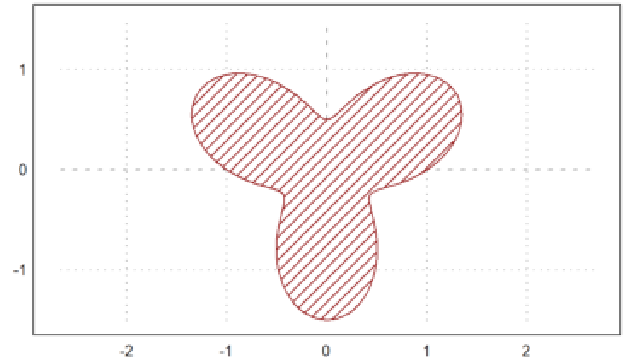
Dalam contoh berikutnya, kami memplot kurva

$$\gamma(t) = (r(t)\cos(t), r(t)\sin(t))$$

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):
```



1
0
-1
-2
-1
0
1
2

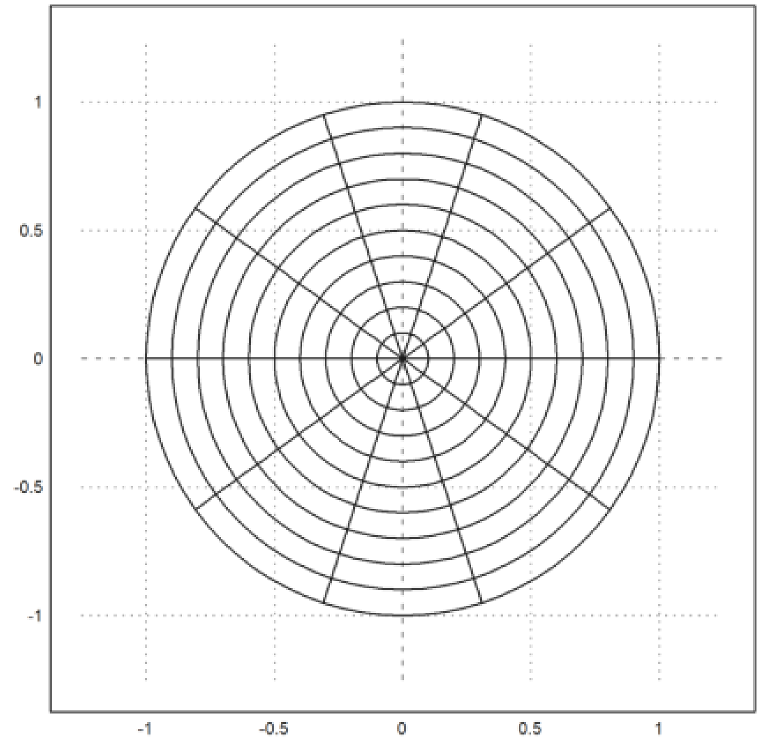
Menggambar Grafik Bilangan Kompleks

Serangkaian bilangan kompleks juga dapat diplot. Kemudian titik-titik grid akan dihubungkan. Jika sejumlah garis kisi ditentukan (atau vektor garis kisi 1x2) dalam argumen cgrid, hanya garis kisi tersebut yang terlihat.

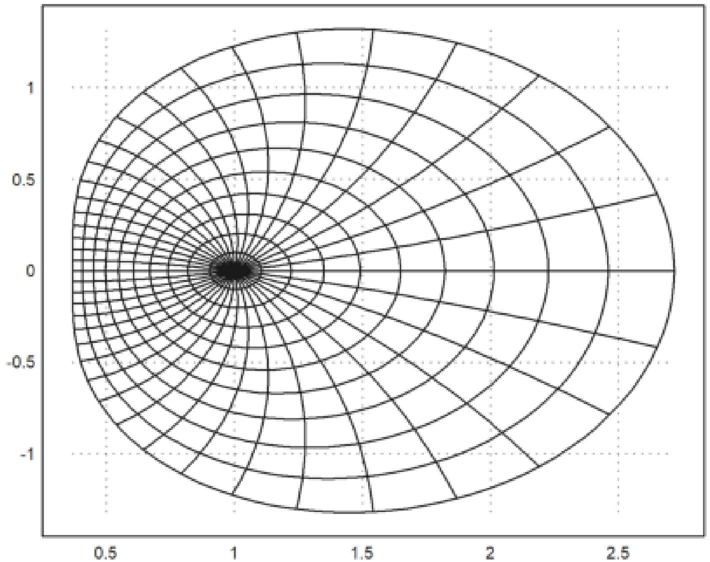
Matriks bilangan kompleks secara otomatis akan diplot sebagai kisi-kisi pada bidang kompleks.

Pada contoh berikut, kita memplot gambar lingkaran satuan di bawah fungsi eksponensial. Parameter cgrid menyembunyikan beberapa kurva grid.

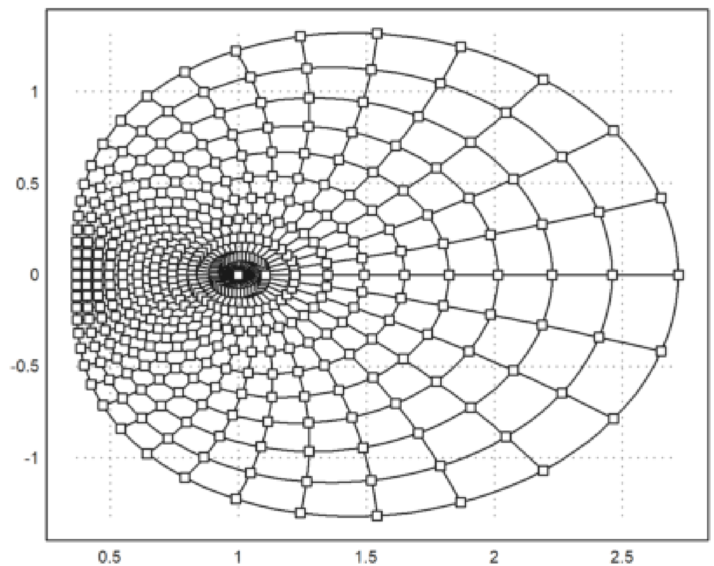
```
>aspect(); r=linspace(0,1,50); a=linspace(0,2pi,80)'; z=r*exp(I*a);...
>plot2d(z,a=-1.25,b=1.25,c=-1.25,d=1.25,cgrid=10):
```




```
>aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)'; z=r*exp(I*a);
>plot2d(exp(z),cgrid=[40,10]):
```



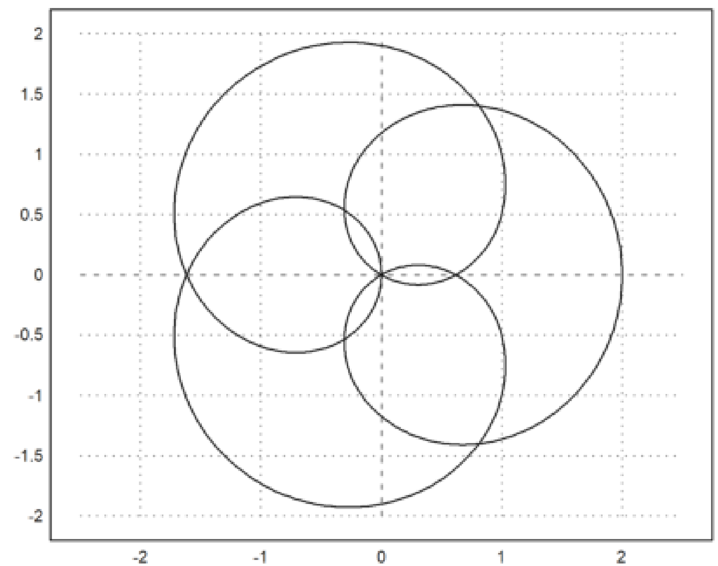

```
>r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=r*exp(I*a);
>plot2d(exp(z),>points,>add):
```



Sebuah vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian real 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):
```


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

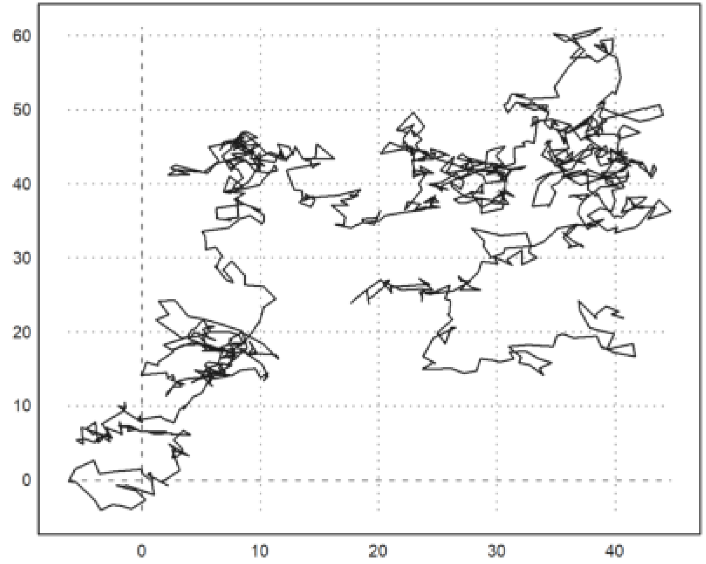
Ada banyak fungsi yang dikhususkan pada plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

Jumlah kumulatif dari nilai terdistribusi normal 0-1 menghasilkan jalan acak.

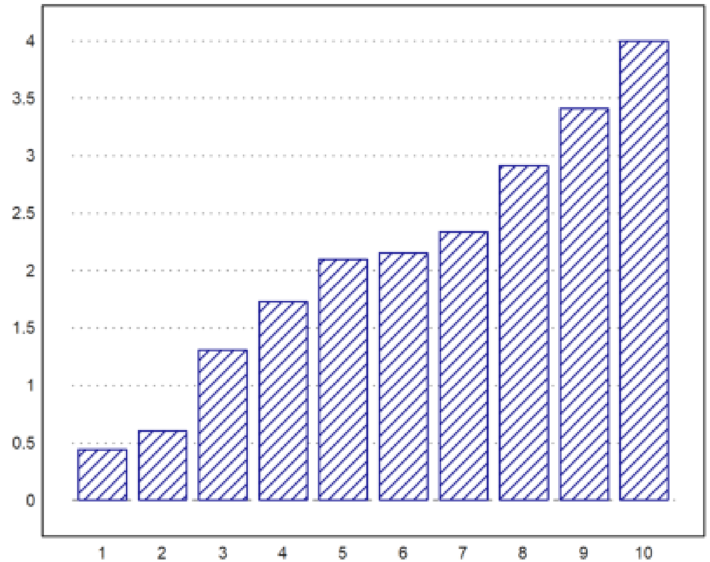
```
>plot2d(cumsum(randnormal(1,1000))):
```

Penggunaan dua baris menunjukkan jalan dalam dua dimensi.

```
>X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]):
```

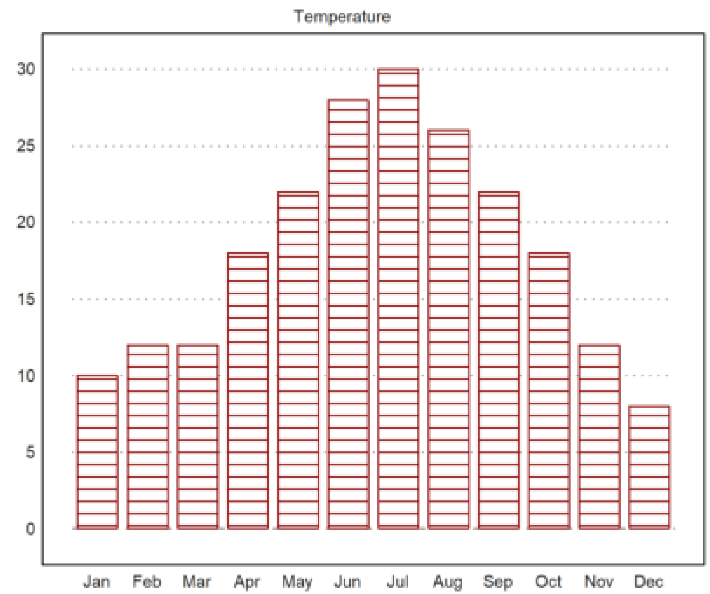



```
>columnsplot(cumsum(random(10)),style="/",color=blue):
```

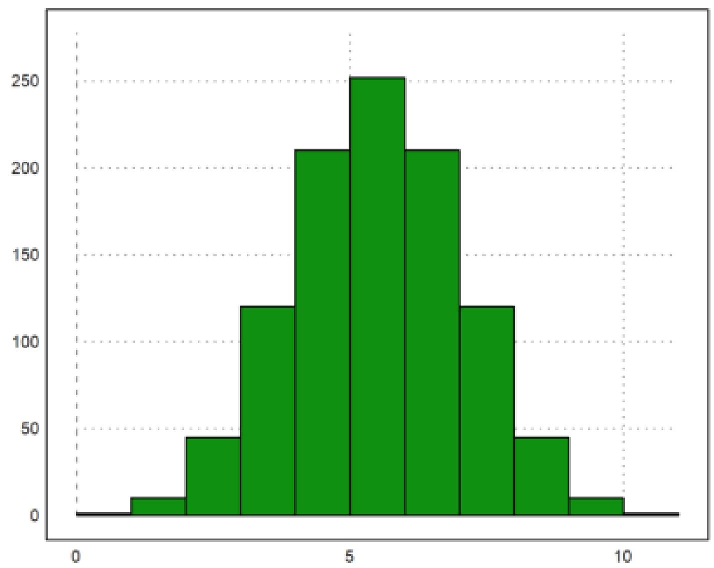


Itu juga dapat menampilkan string sebagai label.

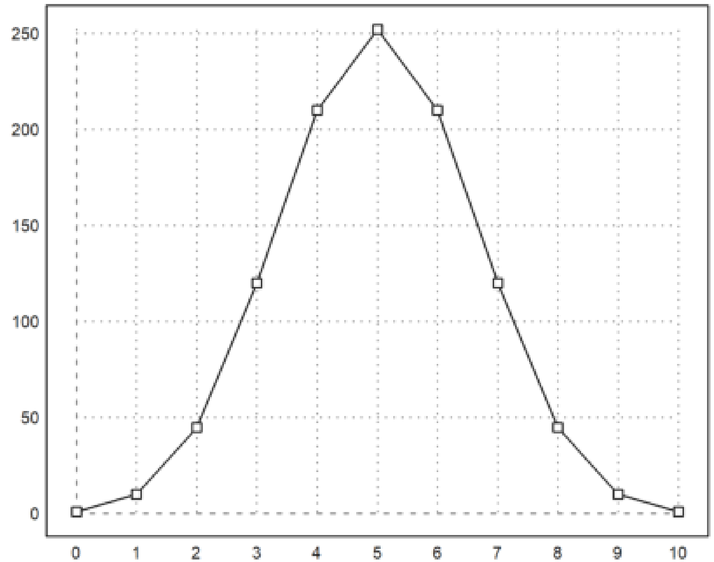
```
>months=["Jan","Feb","Mar","Apr","May","Jun", ...
>  "Jul","Aug","Sep","Oct","Nov","Dec"];
>values=[10,12,12,18,22,28,30,26,22,18,12,8];
>columnsplot(values,lab=months,color=red,style="-");
>title("Temperature"):
```



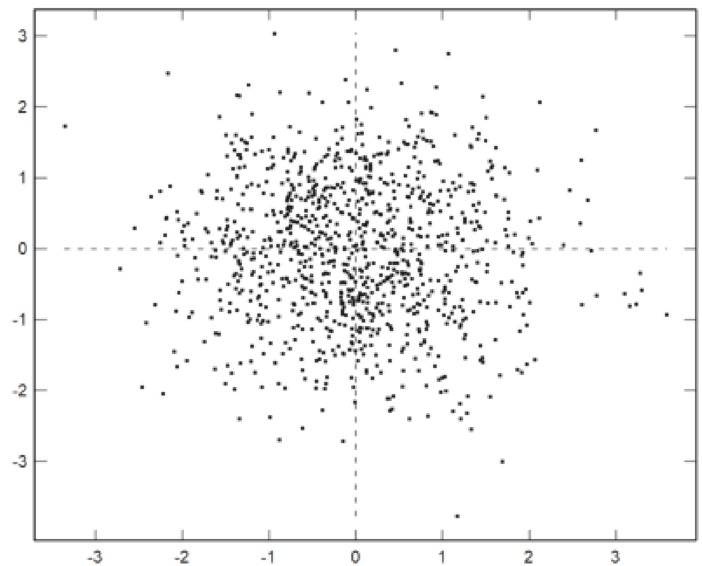

```
>k=0:10;
>plot2d(k, bin(10,k),>bar):
```



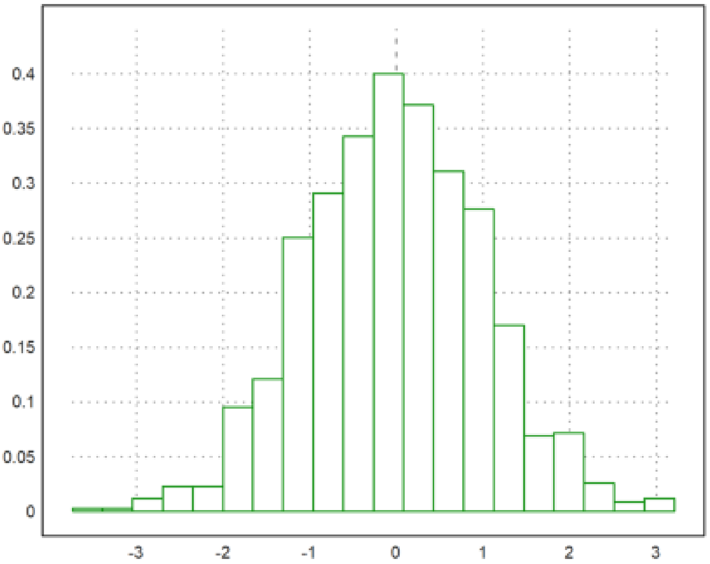

```
>plot2d(k,bin(10,k)); plot2d(k,bin(10,k),>points,>add):
```



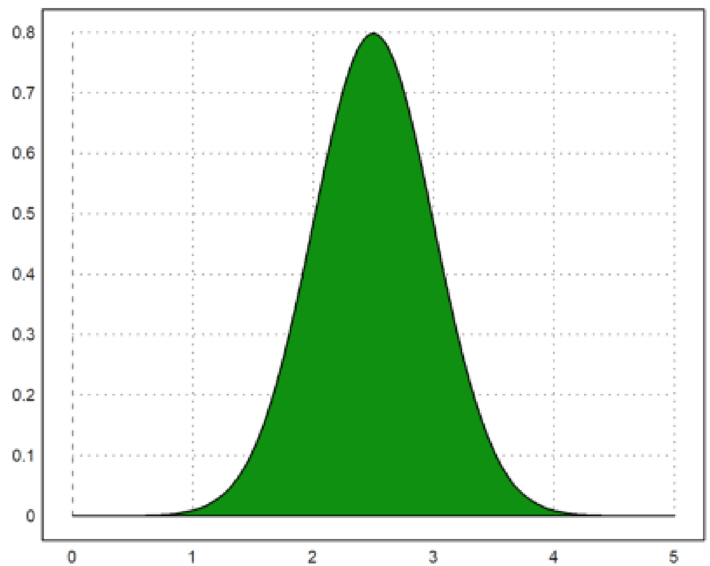

```
>plot2d(normal(1000),normal(1000),>points,grid=6,style=".."):
```




```
>plot2d(normal(1,1000),>distribution,style="O"):
```

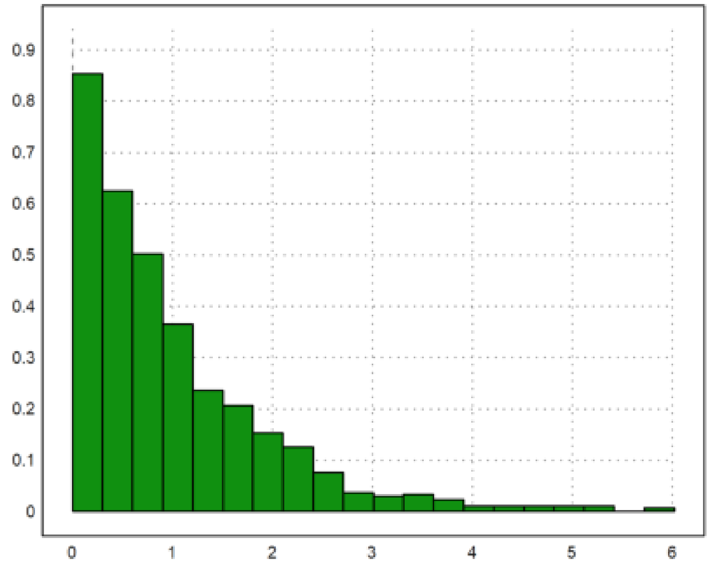



```
>plot2d("qnormal",0,5;2.5,0.5,>filled):
```



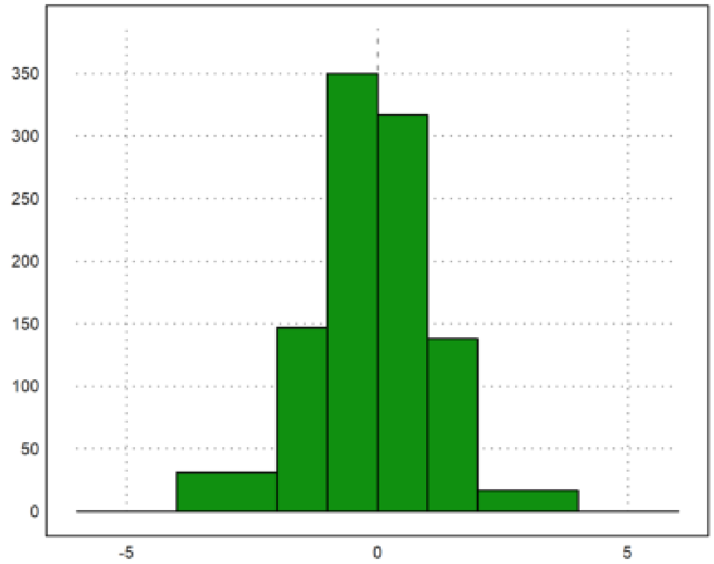
Untuk memplot distribusi statistik eksperimental, Anda dapat menggunakan distribution=n dengan plot2d.

```
>w=randexponential(1,1000); // exponential distribution
>plot2d(w,>distribution): // or distribution=n with n intervals
```



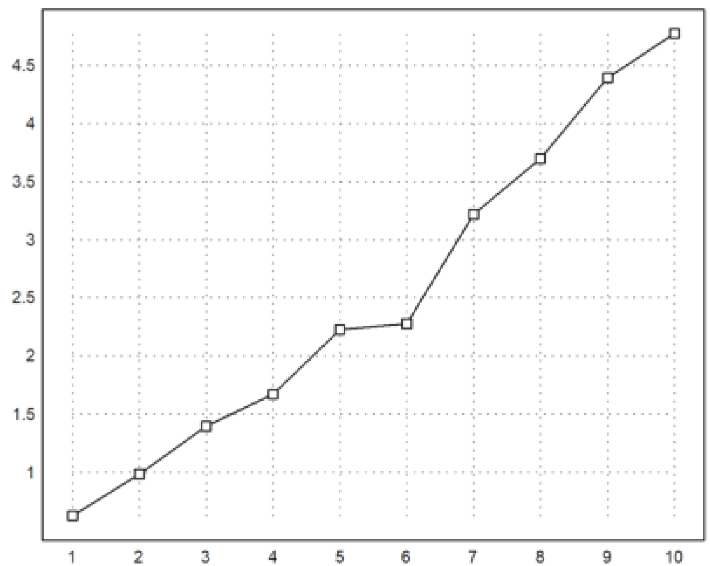
Atau Anda dapat menghitung distribusi dari data dan memplot hasilnya dengan >bar di plot3d, atau dengan plot kolom.

```
>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):
```

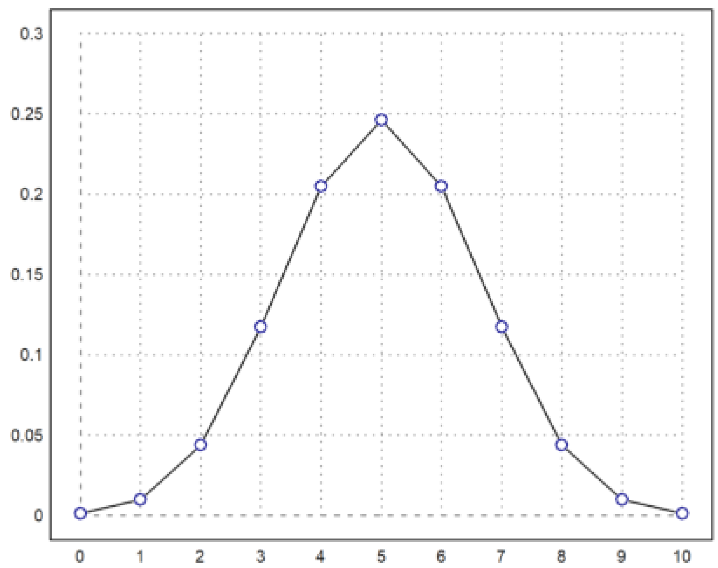


Fungsi statplot() mengatur gaya dengan string sederhana.

```
>statplot(1:10,cumsum(random(10)),"b"):
```



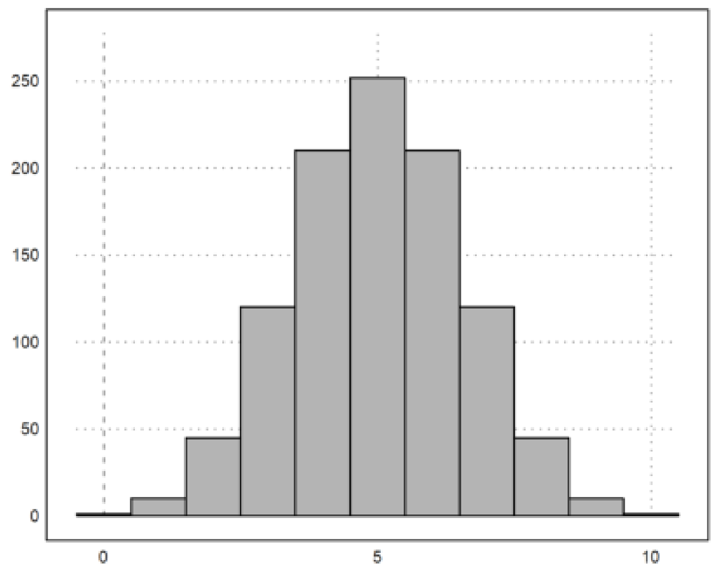

```
>n=10; i=0:n; ...
>plot2d(i,bin(n,i)/2^n,a=0,b=10,c=0,d=0.3); ...
>plot2d(i,bin(n,i)/2^n,points=true,style="ow",add=true,color=blue):
```



Selain itu, data dapat diplot sebagai batang. Dalam hal ini, x harus diurutkan dan satu elemen lebih panjang dari y. Batangnya akan memanjang dari x[i] hingga x[i+1] dengan nilai y[i]. Jika x berukuran sama dengan y, maka x akan diperpanjang satu elemen dengan spasi terakhir.

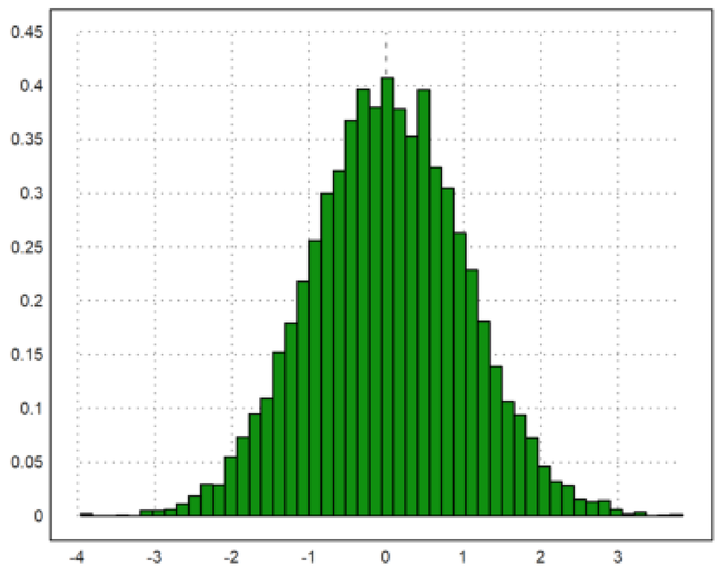
Gaya isian dapat digunakan seperti di atas.

```
>n=10; k=bin(n,0:n); ...
>plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray):
```

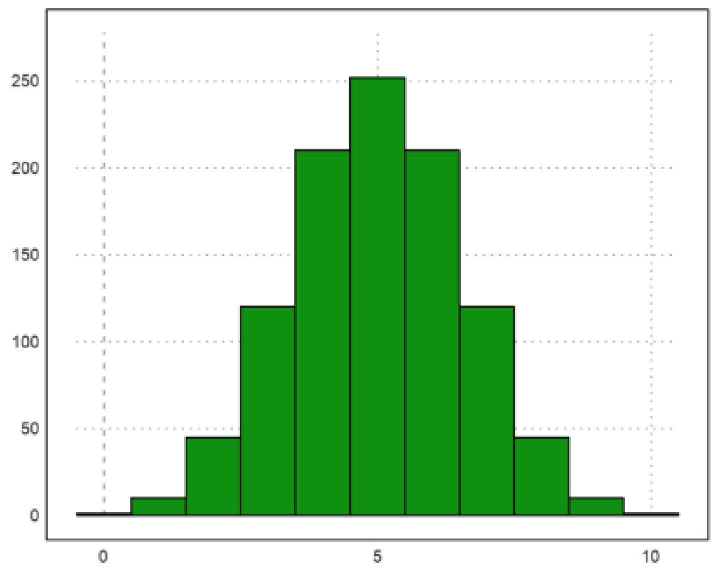


Data untuk plot batang (batang=1) dan histogram (histogram=1) dapat diberikan secara eksplisit dalam xv dan yv, atau dapat dihitung dari distribusi empiris dalam xv dengan >distribusi (atau distribusi=n). Histogram nilai xv akan dihitung secara otomatis dengan >histogram. Jika >even ditentukan, nilai xv akan dihitung dalam interval bilangan bulat.

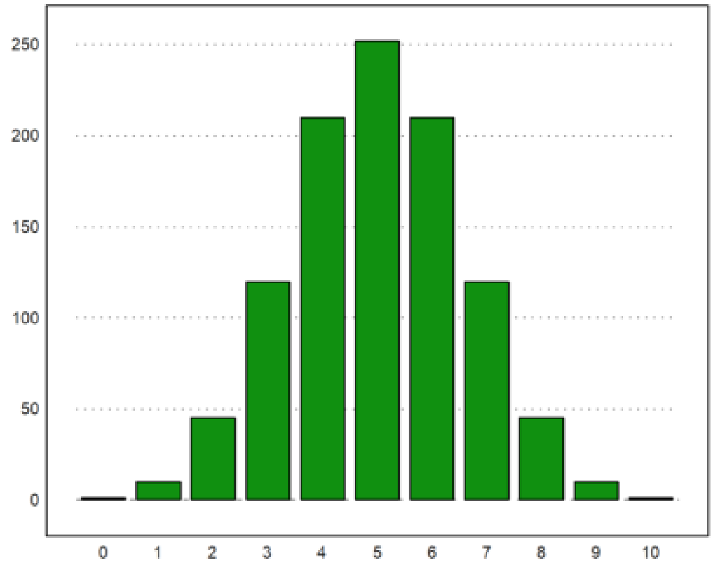
```
>plot2d(normal(10000),distribution=50):
```



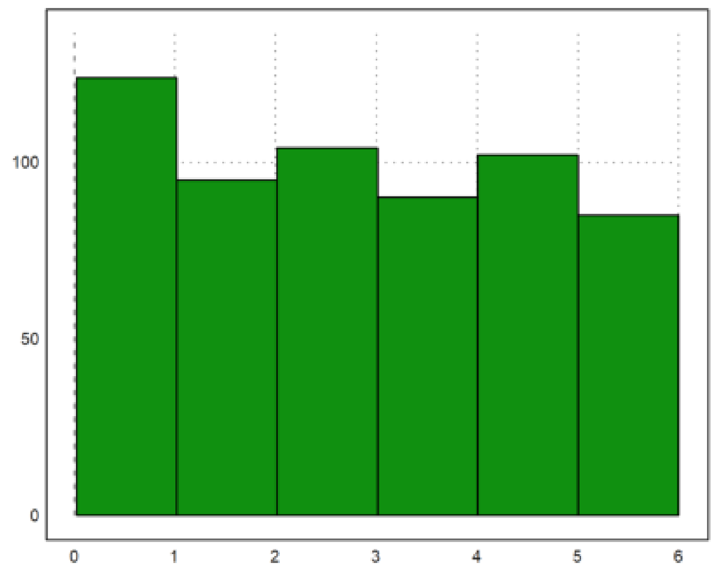

```
>k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m,>bar):
```




```
>columnsplot(m,k):
```

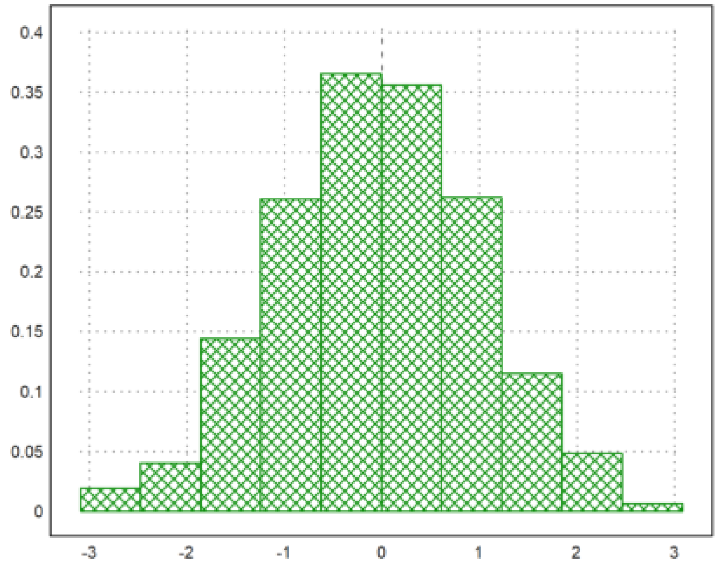



```
>plot2d(random(600)*6,histogram=6):
```



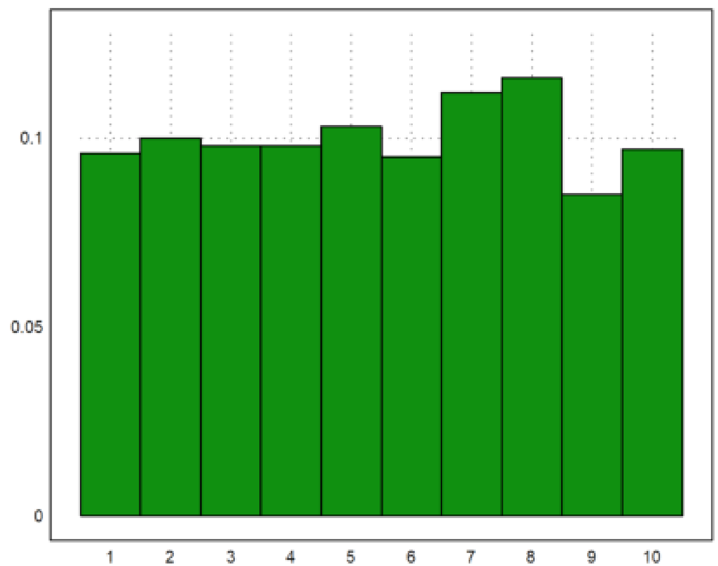
Untuk distribusi, terdapat parameter distribution=n, yang menghitung nilai secara otomatis dan mencetak distribusi relatif dengan n sub-interval.

```
>plot2d(normal(1,1000),distribution=10,style="\/"):
```



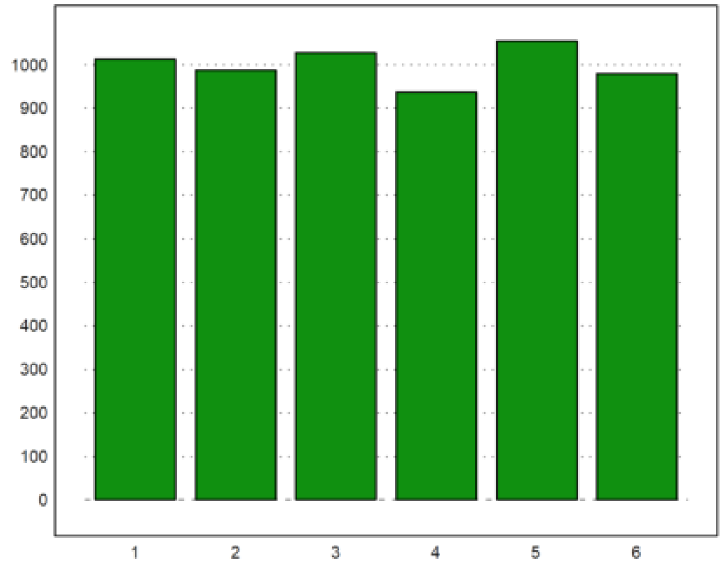
Dengan parameter even=true, ini akan menggunakan interval bilangan bulat.

```
>plot2d(intrandom(1,1000,10),distribution=10,even=true):
```

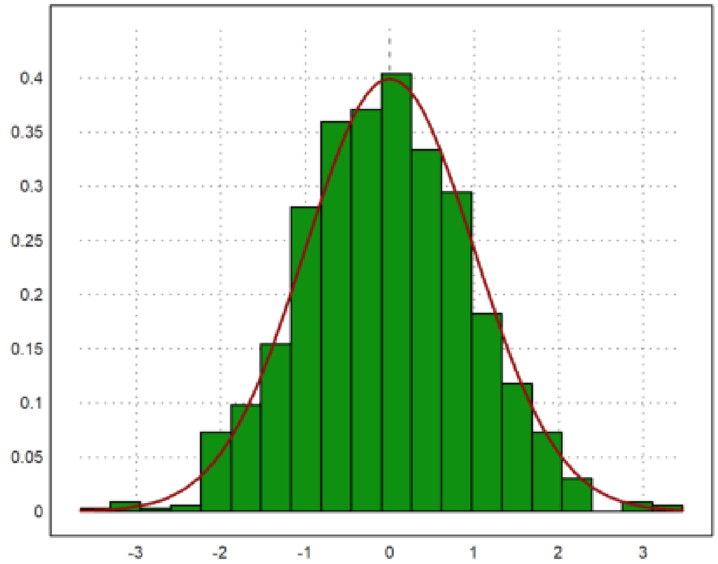


Perhatikan bahwa ada banyak plot statistik yang mungkin berguna. Silahkan lihat tutorial tentang statistik.

```
>columnsplot(getmultiplicities(1:6,intrandom(1,6000,6))):
```

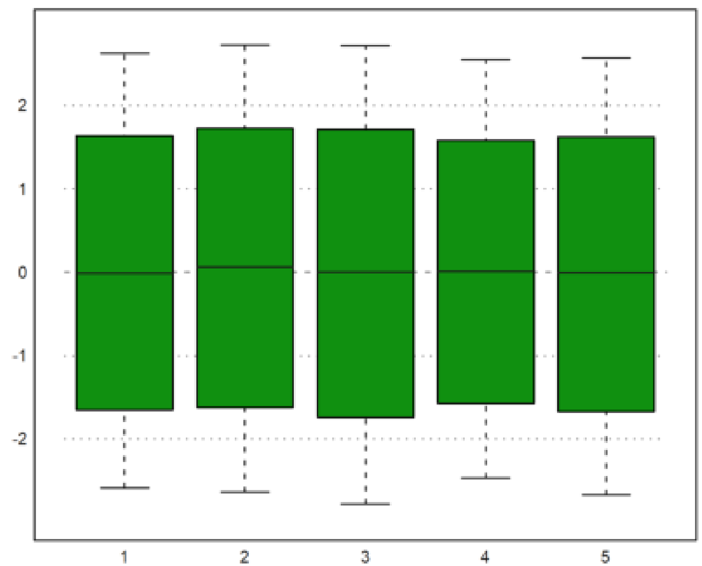



```
>plot2d(normal(1,1000),>distribution); ...
>  plot2d("qnormal(x)",color=red,thickness=2,>add):
```



Ada juga banyak plot khusus untuk statistik. Plot kotak menunjukkan kuartil distribusi ini dan banyak outlier. Menurut definisinya, outlier dalam plot kotak adalah data yang melebihi 1,5 kali rentang 50% tengah plot.

```
>M=normal(5,1000); boxplot(quartiles(M)):
```


2
1
0
-1
-2
1
2
3
4
5

Plot implisit menunjukkan penyelesaian garis level f(x,y)=level, dengan "level" dapat berupa nilai tunggal atau vektor nilai. Jika level = "auto", akan ada garis level nc, yang akan tersebar antara fungsi minimum dan maksimum secara merata. Warna yang lebih gelap atau terang dapat ditambahkan dengan >hue untuk menunjukkan nilai fungsi. Untuk fungsi implisit, xv harus berupa fungsi atau ekspresi parameter x dan y, atau alternatifnya, xv dapat berupa matriks nilai.

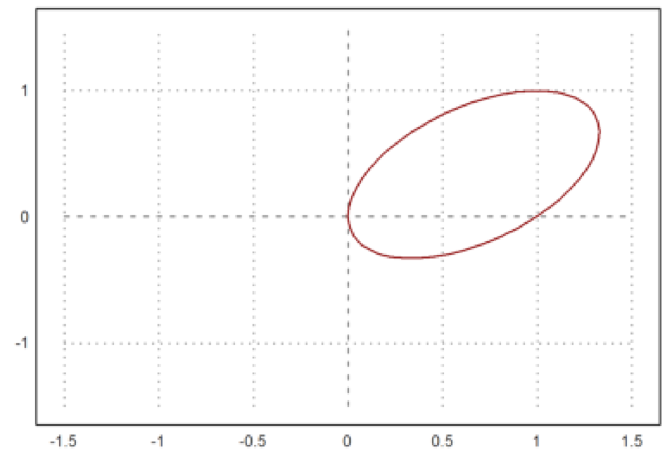
Euler dapat menandai garis level

$$f(x,y) = c$$

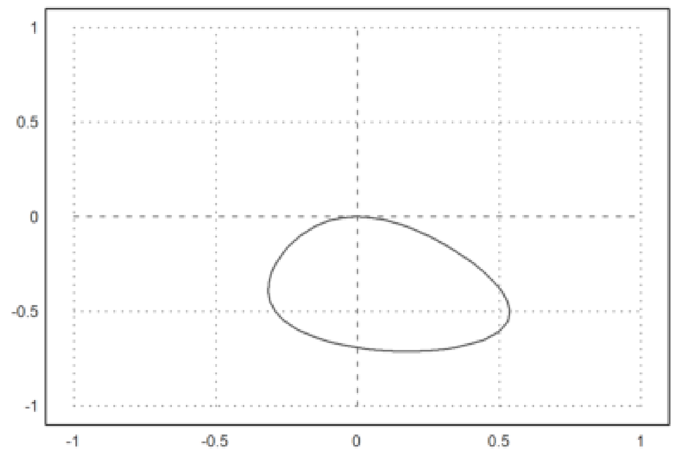
dari fungsi apa pun.

Untuk menggambar himpunan f(x,y)=c untuk satu atau lebih konstanta c, Anda dapat menggunakan plot2d() dengan plot implisitnya pada bidang. Parameter c adalah level=c, dimana c dapat berupa vektor garis level. Selain itu, skema warna dapat digambar di latar belakang untuk menunjukkan nilai fungsi setiap titik dalam plot. Parameter "n" menentukan kehalusan plot.

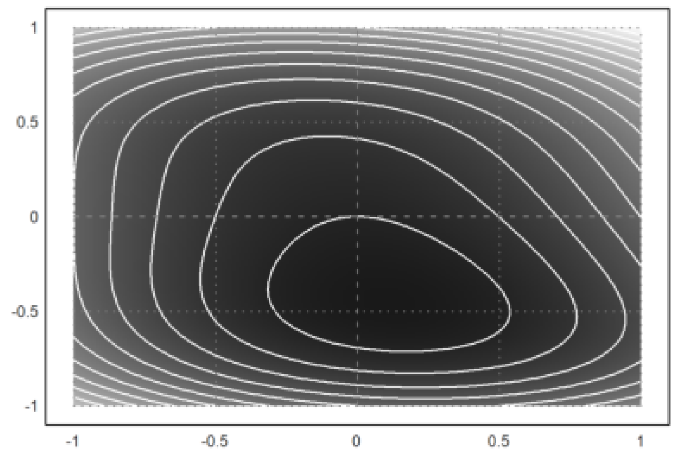
```
>aspect(1.5);
>plot2d("x^2+y^2-x*y-x",r=1.5,level=0,contourcolor=red):
```



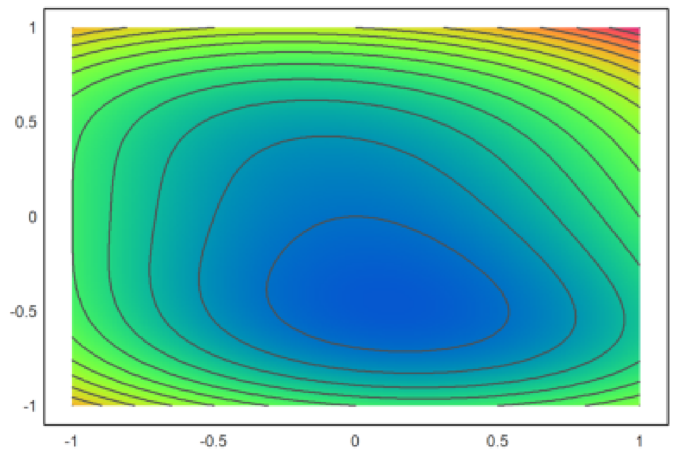

```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr, level=0): // Solutions of f(x,y)=0
```




```
>plot2d(expr,level=0:0.5:20,>hue,contourcolor=white,n=200): // nice
```

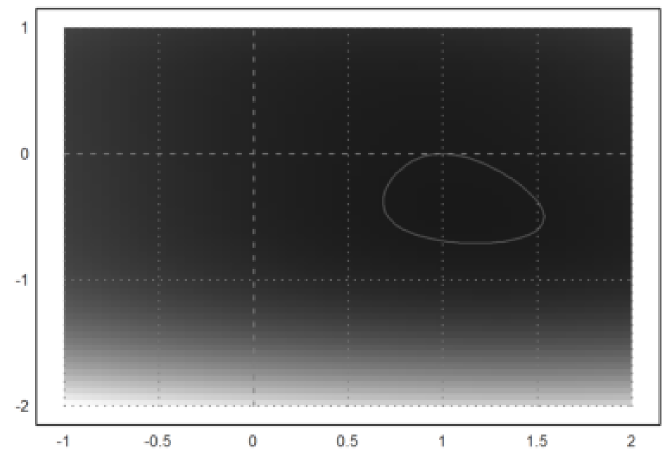



```
>plot2d(expr,level=0:0.5:20,>hue,>spectral,n=200,grid=4): // nicer
```

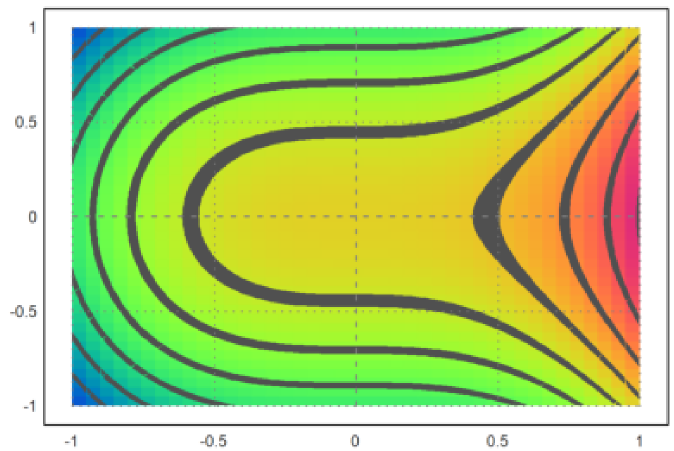


Ini juga berfungsi untuk plot data. Namun Anda harus menentukan rentangnya untuk label sumbu.

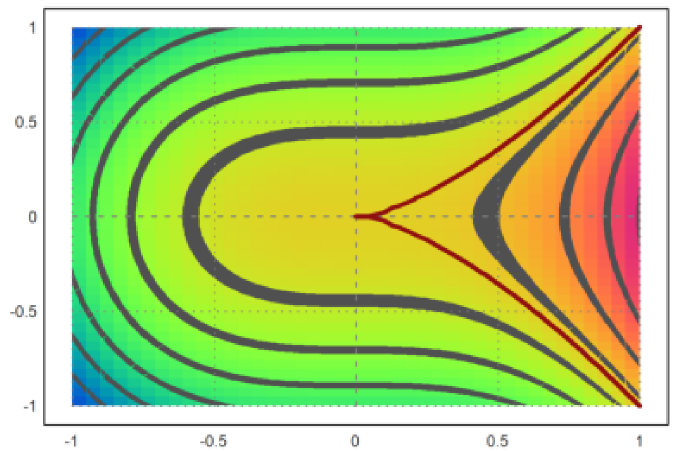
```
>x=-2:0.05:1; y=x'; z=expr(x,y);
>plot2d(z,level=0,a=-1,b=2,c=-2,d=1,>hue):
```



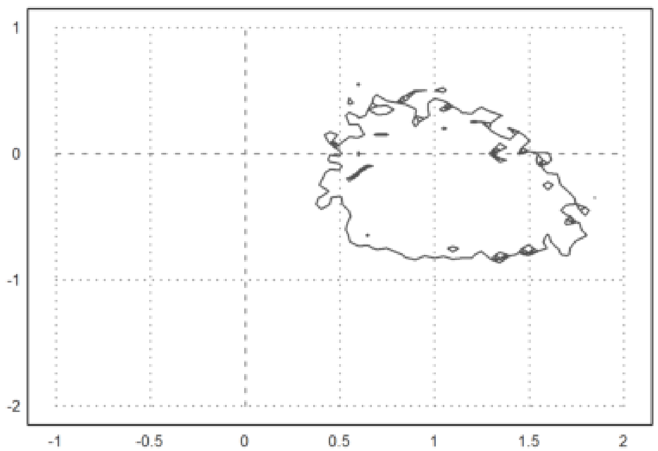

```
>plot2d("x^3-y^2",>contour,>hue,>spectral):
```



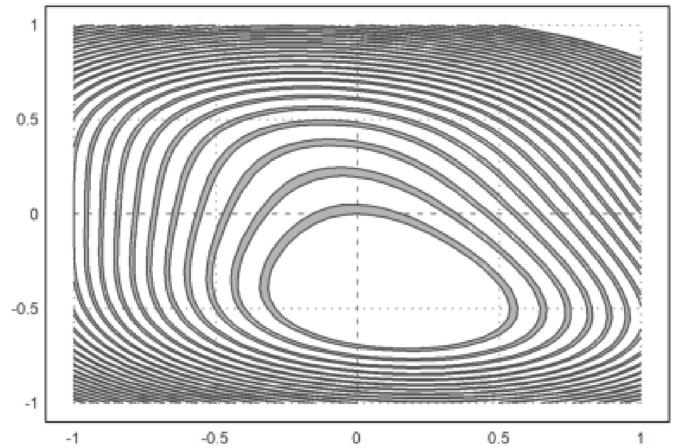

```
>plot2d("x^3-y^2",level=0,contourwidth=3,>add,contourcolor=red):
```



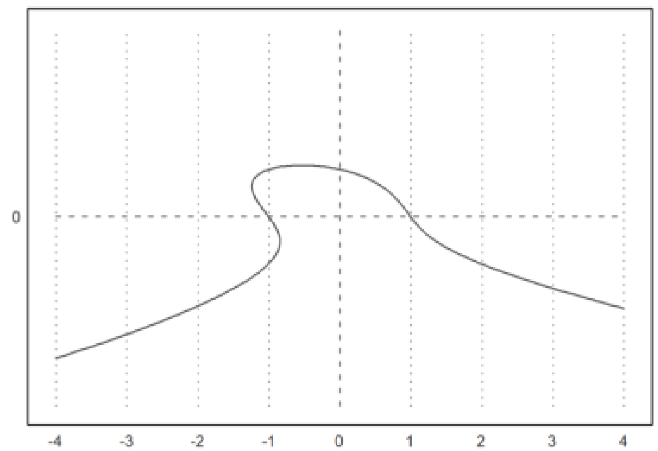

```
>z=z+normal(size(z))*0.2;
>plot2d(z,level=0.5,a=-1,b=2,c=-2,d=1):
```



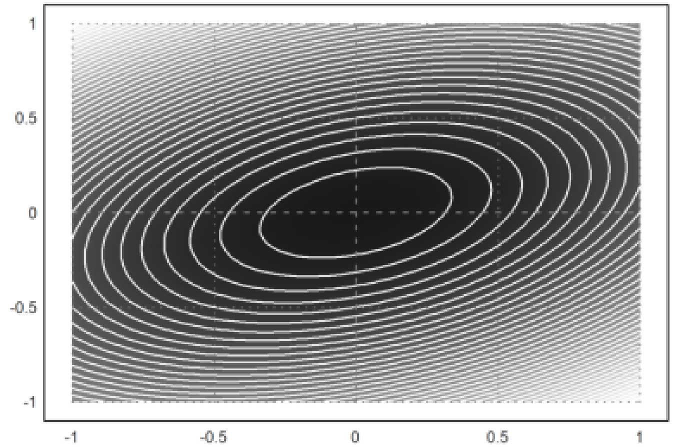

```
>plot2d(expr,level=[0:0.2:5;0.05:0.2:5.05],color=lightgray):
```




```
>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):
```



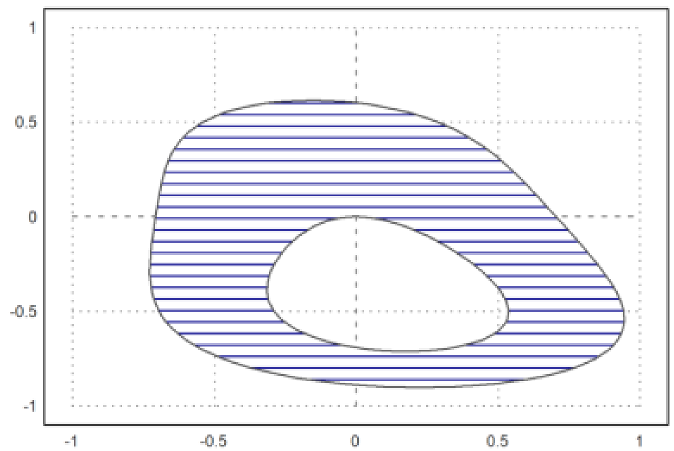
Dimungkinkan juga untuk mengisi set

$$a \leq f(x, y) \leq b$$

dengan rentang level.

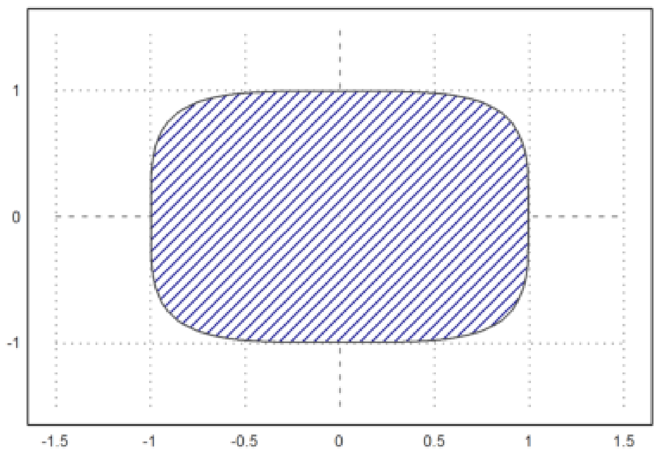
Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks 2xn. Baris pertama adalah batas bawah dan baris kedua berisi batas atas.

```
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <= 1
```

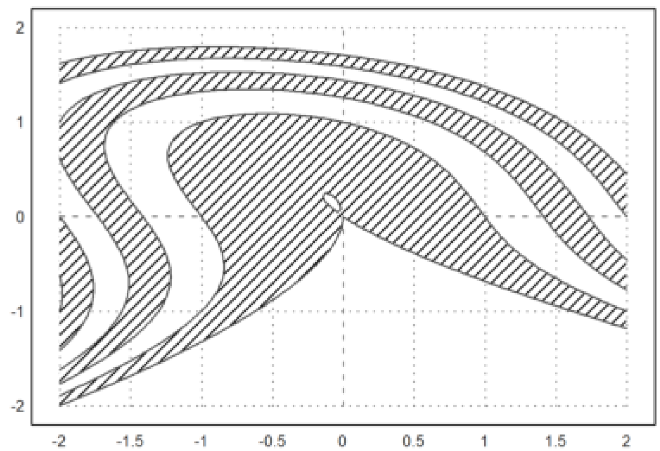


Plot implisit juga dapat menunjukkan rentang level. Maka level harus berupa matriks interval level 2xn, di mana baris pertama berisi awal dan baris kedua berisi akhir setiap interval. Alternatifnya, vektor baris sederhana dapat digunakan untuk level, dan parameter dl memperluas nilai level ke interval.

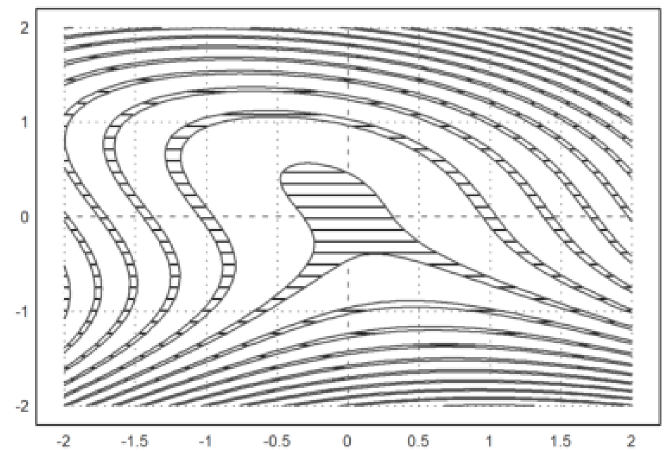
```
>plot2d("x^4+y^4",r=1.5,level=[0;1],color=blue,style="/"):
```



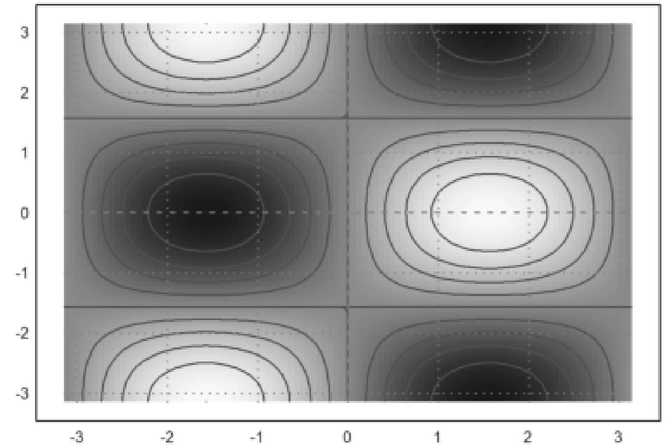

```
>plot2d("x^2+y^3+x*y",level=[0,2,4;1,3,5],style="/",r=2,n=100):
```




```
>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):
```

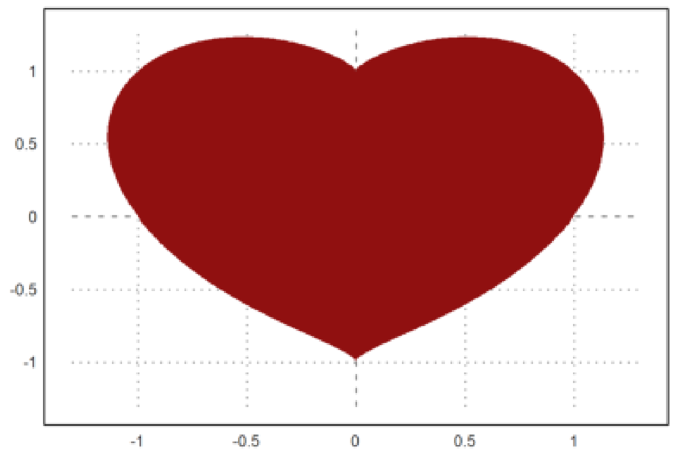


Dimungkinkan juga untuk menandai suatu wilayah

$$a \leq f(x, y) \leq b.$$

Hal ini dilakukan dengan menambahkan level dengan dua baris.

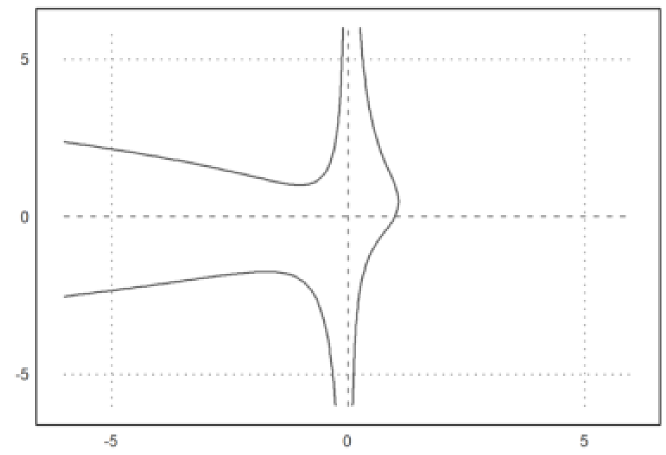
```
>plot2d("(x^2+y^2-1)^3-x^2*y^3",r=1.3, ...
>  style="#",color=red,<outline, ...
>  level=[-2;0],n=100):
```



Dimungkinkan untuk menentukan level tertentu. Misalnya, kita dapat memplot solusi 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) ...
```

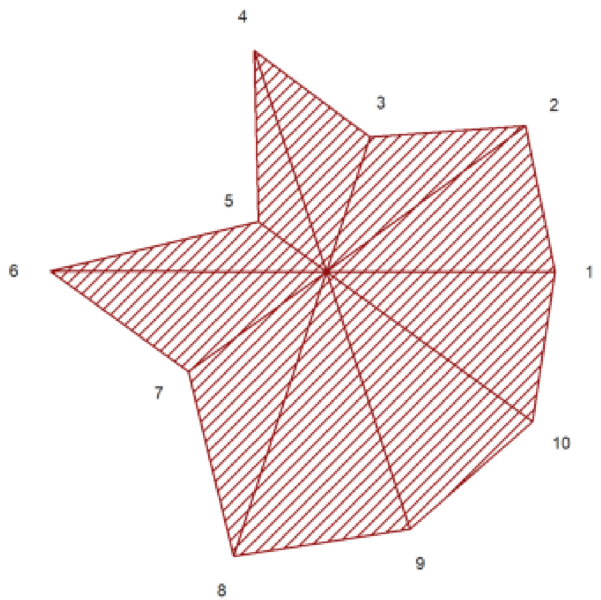
```
if !holding() then clg; endif;
w=window(); window(0,0,1024,1024);
h=holding(1);
r=max(abs(v))*1.2;
setplot(-r,r,-r,r);
n=cols(v); t=linspace(0,2pi,n);
v=v|v[1]; c=v*cos(t); s=v*sin(t);
cl=barcolor(color); st=barstyle(style);
loop 1 to n
  polygon([0,c[#],c[#+1]],[0,s[#],s[#+1]],1);
  if lab!=none 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 tanda centang kotak atau sumbu di sini. Selain itu, kami menggunakan jendela penuh untuk plotnya.

Kami memanggil reset sebelum kami menguji plot ini untuk mengembalikan default grafis. Ini tidak perlu dilakukan jika Anda yakin plot Anda berhasil.

```
>reset; starplot1(normal(1,10)+5,color=red,lab=1:10):
```



Terkadang, Anda mungkin ingin merencanakan sesuatu yang plot2d tidak bisa lakukan, tapi hampir.

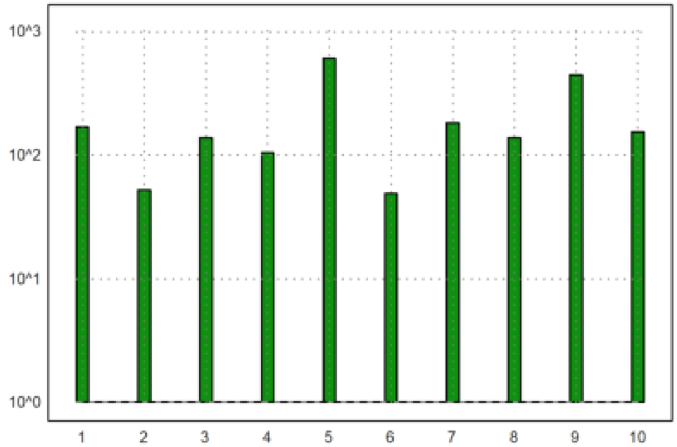
Dalam fungsi berikut, kita membuat plot impuls logaritmik. plot2d dapat melakukan plot logaritmik, tetapi tidak untuk batang impuls.

```
>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 menganimasikan kurva 2D menggunakan plot langsung. Perintah plot(x,y) hanya memplot kurva ke dalam jendela plot. setplot(a,b,c,d) menyetel jendela ini.

Fungsi wait(0) memaksa plot muncul di jendela grafis. Jika tidak, pengundian ulang akan 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
```

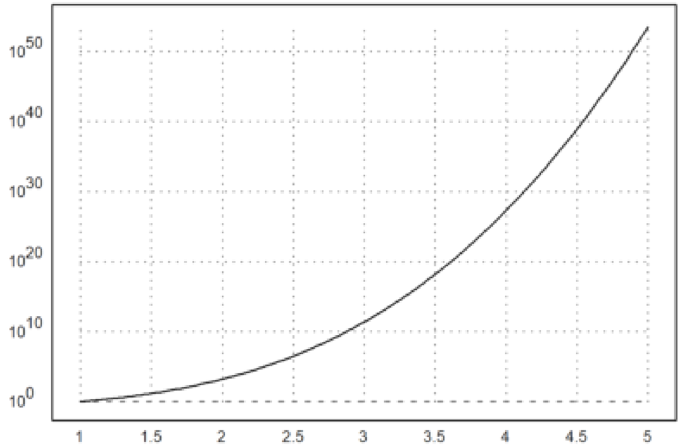
Tekan tombol apa saja untuk menghentikan animasi ini.

```
>animliss(2,3); // lihat hasilnya, jika sudah puas, tekan ENTER
```

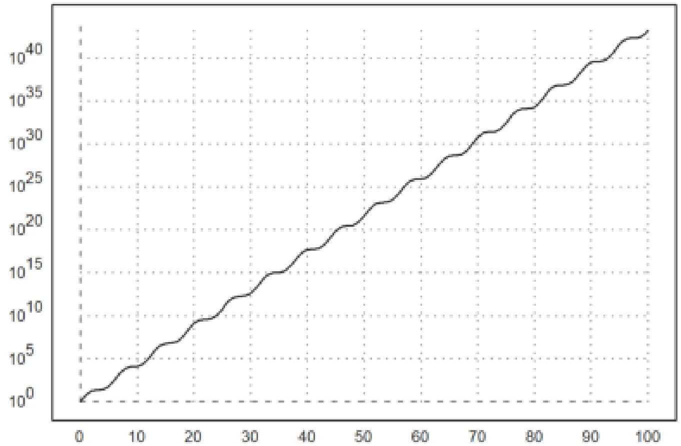
EMT menggunakan parameter ”logplot” untuk skala logaritmik.
Plot logaritma dapat diplot menggunakan skala logaritma di y dengan logplot=1, atau menggunakan skala logaritma di x dan y dengan logplot=2, atau di x dengan logplot=3.

```
 - logplot=1: y-logaritma
 - logplot=2: x-y-logaritma
 - logplot=3: x-logaritma
```

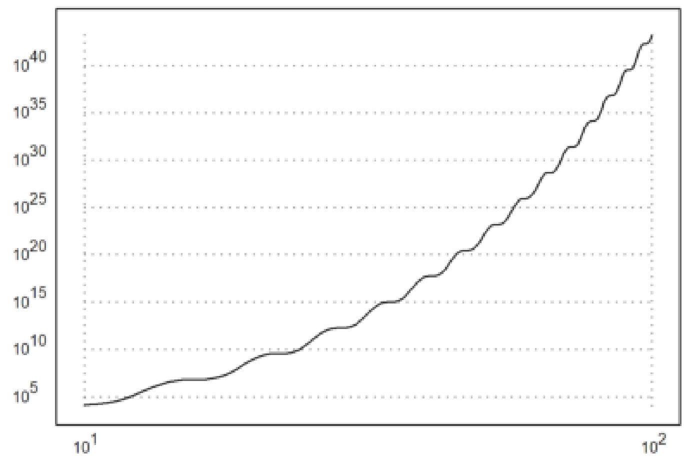
```
>plot2d("exp(x^3-x)*x^2",1,5,logplot=1):
```



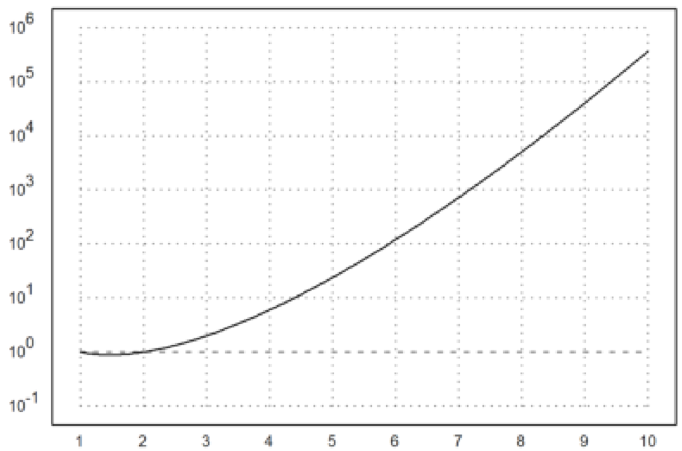
```
>plot2d("exp(x+sin(x))",0,100,logplot=1):
```



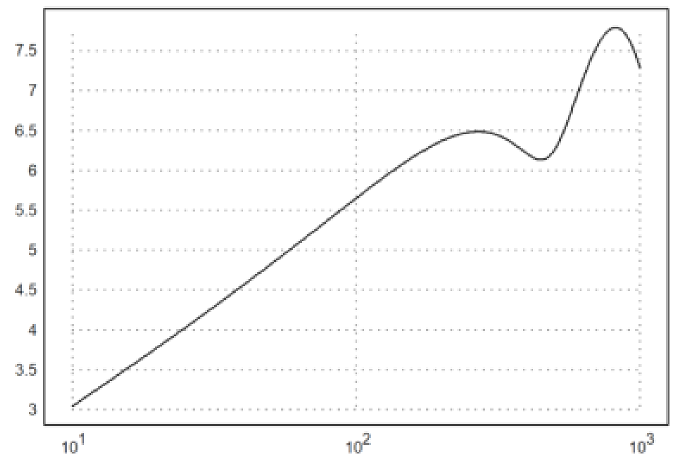

```
>plot2d("exp(x+sin(x))",10,100,logplot=2):
```




```
>plot2d("gamma(x)",1,10,logplot=1):
```

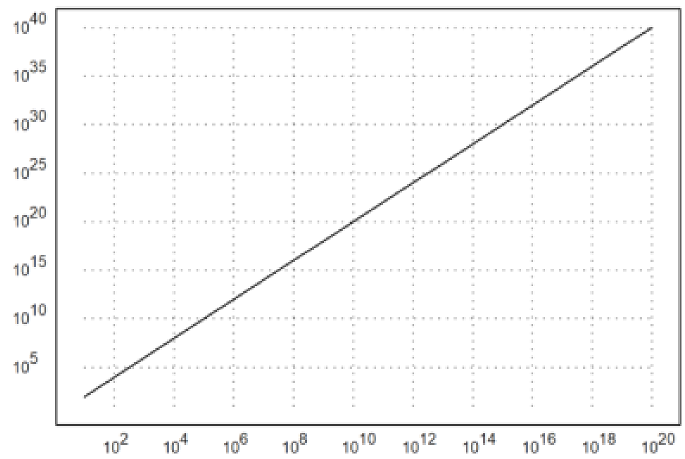



```
>plot2d("log(x*(2+sin(x/100)))",10,1000,logplot=3):
```



Ini juga berfungsi dengan plot data.

```
>x=10^(1:20); y=x^2-x;
>plot2d(x,y,logplot=2):
```



Contoh Latihan Soal

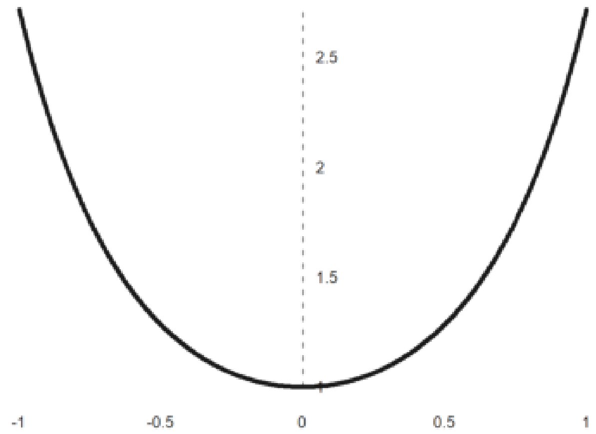
1. Buatlah plot 2 dimensi dari fungsi berikut

$$y = a * (x^2)/a$$

dengan a=4 dan interval dari x=-1 sampai x=1. Kemudian buatlah plot tersebut dalam grid 3 dan ketebalan 3!

Penyelesaian:

```
>a:=4; expr &= exp(a*(x^2)/a);
>plot2d(expr,-1,1,grid=3,thickness=3):
```



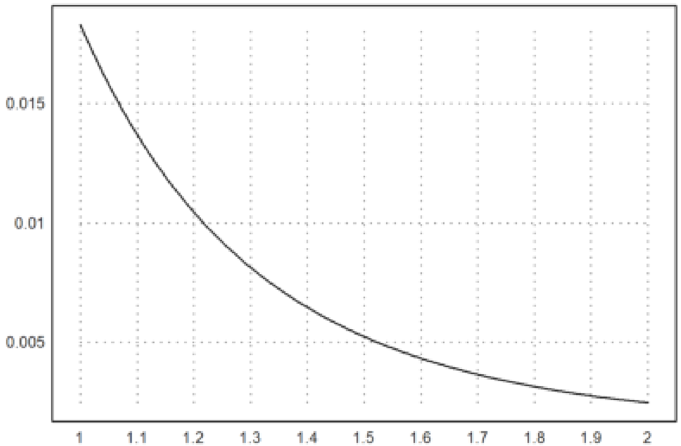
2. Buatlah plot 2 dimensi dari fungsi berikut

$$y(x, a) = x^2 - a * x$$

dengan nilai parameter adalah 5, dengan interval x nya dari 1 sampai 2

Penyelesaian:

```
>a:=5; expr &= exp((x^2)-a*x);
>plot2d(expr,1,2):
```

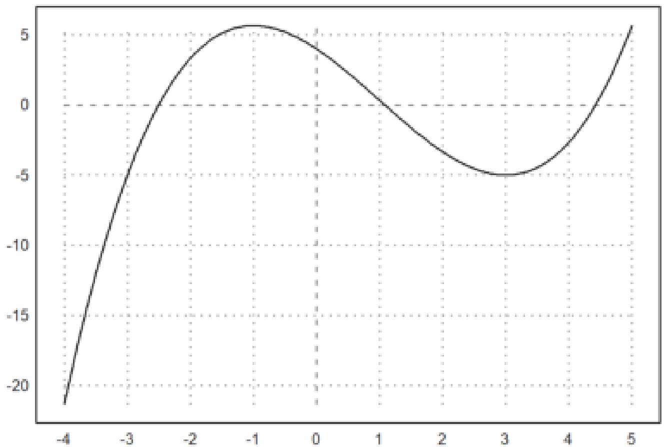


3. Selidiki dimanakah fungsi f(x) berikut naik dan turun

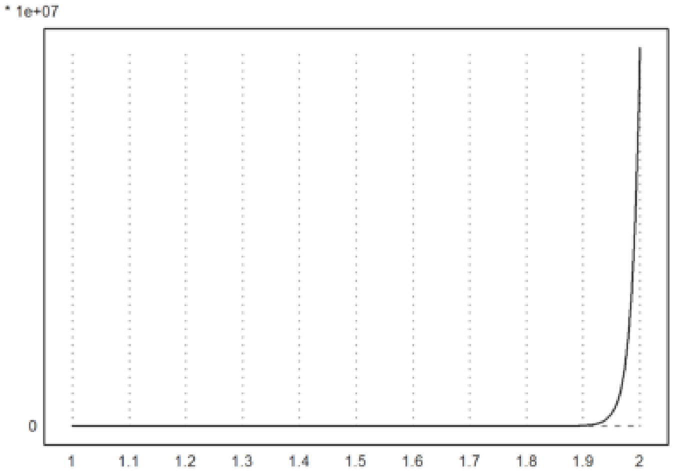
$$f(x) = (1/3)x^3 * -x^2 - 3 * x + 4$$

Penyelesaian:

```
>function f(x):=(1/3)*x^3-x^2-3*x+4;
>plot2d("f",-4,5):
```




```
>a:=8; expr &= exp((x^5)-a*x);
>plot2d(expr,1,2):
```



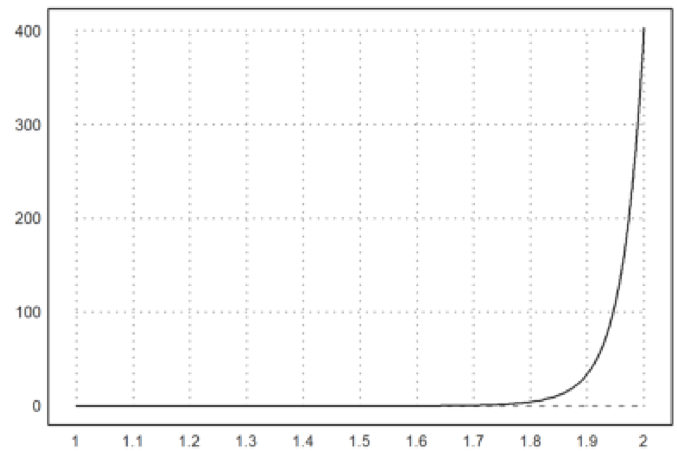
5. Buatlah plot 2 dimensi dari fungsi berikut

$$y(x,a) = x^4 - a * x$$

dengan nilai parameter adalah 5, dengan interval x nya dari 1 sampai 2

Penyelesaian:

```
>a:=5; expr &= exp((x^4)-a*x);
>plot2d(expr,1,2):
```



Rujukan Lengkap Fungsi

plot2d() function plot2d (xv, yv, btest, a, b, c, d, xmin, xmax, r, n, ..
logplot, kisi, bingkai, warna bingkai, kotak, warna, ketebalan, gaya, ..
otomatis, tambahkan, pengguna, delta, poin, titik tambahan, gaya titik, bilah, histogram, ..
distribusi, genap, langkah, sendiri, adaptif, rona, level, kontur, ..
nc, terisi, fillcolor, outline, title, xl, yl, maps, contourcolor, ..
contourwidth, ticks, margin, clipping, cx, cy, insimg, spectral, ..
cgrid, vertikal, lebih kecil, dl, niveau, level)
Fungsi plot serbaguna untuk plot di pesawat (plot 2D). Fungsi ini dapat melakukan
plot fungsi satu variabel, plot data, kurva bidang, plot batang, kisi-kisi bilangan kompleks, dan plot
implisit fungsi dua variabel.
Parameter
x,y : persamaan, fungsi atau vektor data
a,b,c,d : Area plot (default a=-2,b=2)
r : jika r diset, maka a=cx-r, b=cx+r, c=cy-r, d=cy+r
r dapat berupa vektor [rx,ry] atau vektor
rx1,rx2,ry1,ry2
.
xmin,xmax : rentang parameter untuk kurva
auto : Menentukan y-range secara otomatis (default)
kuadrat : jika benar, coba pertahankan rentang x-y persegi
n : jumlah interval (default adaptif)
kisi : 0 = tidak ada kisi dan label,
1 = sumbu saja,
2 = grid normal (lihat di bawah untuk jumlah garis
grid)
3 = sumbu dalam
4 = tidak ada kisi-kisi
5 = kisi penuh termasuk margin
6 = kutu di bingkai
7 = sumbu saja
8 = sumbu saja, sub-centang
bingkai : 0 = tanpa bingkai
framecolor : warna bingkai dan kisi
margin : angka antara 0 dan 0,4 untuk margin di sekitar plot
warna : Warna kurva. Jika ini adalah vektor warna,
itu akan digunakan untuk setiap baris matriks plot. Dalam

kasus plot titik, itu harus berupa vektor kolom. Jika vektor baris atau matriks penuh warna digunakan untuk plot titik, itu akan digunakan untuk setiap titik data.
ketebalan: ketebalan garis untuk kurva
Nilai ini bisa lebih kecil dari 1 untuk garis yang sangat
tipis.
style : Plot style untuk garis, spidol, dan isian.
Untuk poin gunakan
”[]”, ”<>”, ”.”, ”..”, ”...”,
”*”, ”+”, ”|”, ”-”, ”o”
”[]”, ”<>”, ”o” (bentuk terisi)
”[]w”, ”<>w”, ”ow” (tidak transparan)
Untuk penggunaan garis
”-”, ”–”, ”-.”, ”.”, ”.-.”, ”-.-”, ”->”
Untuk poligon terisi atau plot batang gunakan
””, ”O”, ”O”, ”/”, ”\”, ”\/”,
”+”, ”|”, ”-”, ”t”
poin : plot titik tunggal alih-alih segmen garis
addpoints : jika benar, plot segmen garis dan titik
add : menambahkan plot ke plot yang ada
pengguna : aktifkan interaksi pengguna untuk fungsi
delta : ukuran langkah untuk interaksi pengguna
bar : plot batang (x adalah batas interval, y nilai interval)
histogram : memplot frekuensi x dalam n subinterval
distribusi=n : memplot distribusi x dengan n subinterval
even : gunakan nilai antar untuk histogram otomatis.
langkah : memplot fungsi sebagai fungsi langkah (langkah=1,2)
adaptif : gunakan plot adaptif (n adalah jumlah langkah minimal)
level : plot garis level dari fungsi implisit dua variabel
outline : menggambar batas rentang level.
Jika nilai level adalah matriks 2xn, rentang level akan ditarik
dalam warna menggunakan gaya isian yang diberikan. Jika garis besar benar, itu
akan digambar dalam warna kontur. Dengan menggunakan fitur ini, wilayah
f(x,y) antara batas dapat ditandai.
hue : tambahkan warna hue ke plot level untuk menunjukkan fungsinya
nilai

kontur : Gunakan plot level dengan level otomatis
nc : jumlah garis level otomatis
judul : judul plot (default ””)
xl, yl : label untuk sumbu x dan y
lebih kecil : jika >0, akan ada lebih banyak ruang di sebelah kiri untuk label.
vertikal :
Mengaktifkan atau menonaktifkan label vertikal. Ini mengubah
variabel global
verticallabels secara lokal untuk satu plot. Nilai 1 hanya set
vertikal
teks, nilai 2 menggunakan label numerik vertikal pada sumbu y.
terisi : mengisi plot kurva
fillcolor : mengisi warna untuk bar dan kurva yang terisi
outline : batas poligon yang terisi
logplot : mengatur plot logaritma
1 = logplot di y,
2 = plot log di xy,
3 = logplot dalam x
memiliki :
Sebuah string, yang menunjuk ke rutinitas plot sendiri. Dengan >
pengguna, Anda mendapatkan
interaksi pengguna yang sama seperti di plot2d. Rentang akan diatur
sebelum setiap panggilan ke fungsi Anda.
peta : ekspresi peta (0 lebih cepat), fungsi selalu dipetakan.
contourcolor : warna garis kontur
contourwidth : lebar garis kontur
clipping : mengaktifkan clipping (default adalah true)
judul :
Ini dapat digunakan untuk menggambarkan plot. Judul akan muncul di
atas
jalan cerita. Selain itu, label untuk sumbu x dan y dapat
ditambahkan dengan
xl=”string” atau yl=”string”. Label lain dapat ditambahkan dengan
fungsi label() atau labelbox(). Judulnya bisa unicode
string atau gambar rumus Lateks.

jaringan :
Menentukan jumlah garis grid untuk plot grid yang kompleks.
Harus merupakan pembagi dari ukuran matriks dikurangi 1 (jumlah
subinterval). cgrid dapat berupa vektor [cx,cy].
Ringkasan
Fungsi dapat merencanakan
- ekspresi, koleksi panggilan atau fungsi dari satu variabel,
- kurva parametrik,
- data x terhadap data y,
- fungsi implisit,
- petak batang,
- jaringan kompleks,
- poligon.
Jika fungsi atau ekspresi untuk xv diberikan, plot2d() akan
menghitung
nilai dalam rentang yang diberikan menggunakan fungsi atau ekspresi. Itu
ekspresi harus berupa ekspresi dalam variabel x. Rentang harus
didefinisikan dalam parameter a dan b kecuali rentang default
harus digunakan. Rentang y akan dihitung secara otomatis,
kecuali c dan d ditentukan, atau radius r, yang menghasilkan kisaran
r,r
untuk x dan y. Untuk plot fungsi, plot2d akan menggunakan
evaluasi adaptif fungsi secara default. Untuk mempercepat
plot untuk fungsi yang rumit, matikan ini dengan <adaptif, dan
opsional mengurangi jumlah interval n. Selain itu, plot2d()
akan secara default menggunakan pemetaan. Yaitu, itu akan menghitung elemen plot
untuk elemen. Jika ekspresi atau fungsi Anda dapat menangani a
vector x, Anda dapat menonaktifkannya dengan <maps untuk evaluasi yang lebih cepat.
Perhatikan bahwa plot adaptif selalu dihitung elemen untuk elemen.
Jika fungsi atau ekspresi untuk xv dan untuk yv ditentukan,
plot2d() akan menghitung kurva dengan nilai xv sebagai koordinat x
dan nilai yv sebagai koordinat y. Dalam hal ini, rentang harus
didefinisikan untuk parameter menggunakan xmin, xmax. Ekspresi yang terkandung
dalam string harus selalu ekspresi dalam variabel parameter x.

Menggambar Plot 3D dengan EMT

Ini adalah pengenalan plot 3D di Euler. Kita membutuhkan plot 3D untuk memvisualisasikan fungsi dari dua variabel.

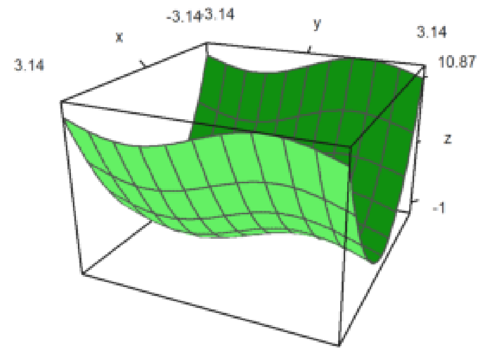
Euler menggambar fungsi tersebut menggunakan algoritma pengurutan untuk menyembunyikan bagian di latar belakang. Secara umum, Euler menggunakan proyeksi pusat. Standarnya adalah dari kuadran x-y positif menuju titik asal x=y=z=0, tetapi sudut=0° terlihat dari arah sumbu y. Sudut pandang dan tinggi dapat diubah.

Euler dapat merencanakan

- permukaan dengan bayangan dan garis level atau rentang level,
- awan poin,
- kurva parametrik,
- permukaan implisit.

Plot 3D dari suatu fungsi menggunakan plot3d. Cara termudah adalah dengan memplot ekspresi dalam x dan y. Parameter r mengatur kisaran plot di sekitar (0,0).

```
>aspect(1.5); plot3d("x^2+sin(y)",r=pi):
```



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. Biasanya, grafik ini digunakan untuk menggambarkan hubungan antara dua variabel dalam bidang dua dimensi dan tiga dimensi. Dalam konteks ini, variabel independen (x dan y) adalah variabel input, sedangkan variabel dependen (z) adalah variabel output yang dihasilkan oleh ekspresi matematis.

Rumus umum untuk menggambar grafik fungsi dua variabel dalam bentuk ekspresi langsung adalah:

$$z = f(x, y)$$

Dalam rumus ini:
- z adalah variabel dependen yang ingin kita gambar dalam grafik.
- f(x, y) adalah ekspresi matematis yang menghubungkan variabel z dengan variabel independen x dan y. Ekspresi ini dapat berupa fungsi linear, fungsi kuadrat, fungsi akar kuadrat, eksponensial, logaritma, trigonometri, nilai mutlak, atau jenis fungsi matematis lainnya, tergantung pada hubungan yang ingin diilustrasikan.

1. Grafik Fungsi Linear

Fungsi linear dua variabel biasanya dinyatakan dalam bentuk

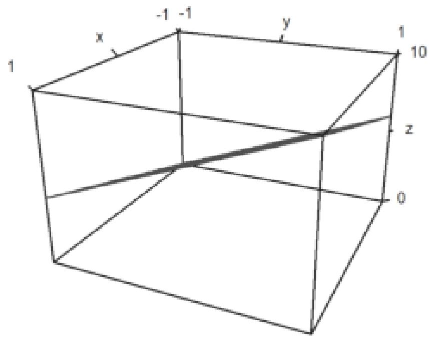
$$f(x, y) = ax + by + c$$

dimana a,b, dan c adalah konstanta. Grafik fungsi linear ini adalah sebuah bidang datar, dan bentuknya akan bervariasi tergantung pada nilai a dan b.

Contoh:

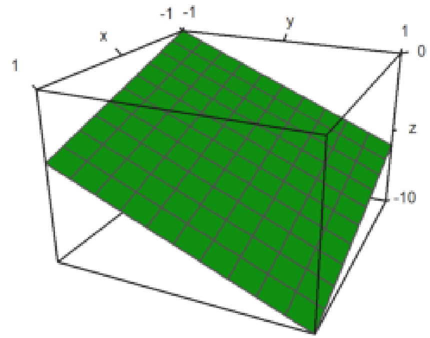
$$f(x, y) = 2x + 3y + 5$$

```
>plot3d("2*x+3*y+5"):
```



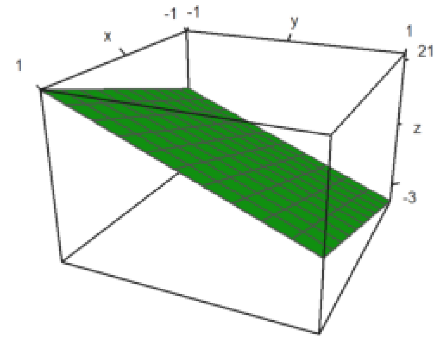
$$f(x, y) = -2x - 3y - 5$$

```
>plot3d("-2*x-3*y-5"):
```



$$f(x,y) = 5x - 7y + 9$$

```
>plot3d("5*x-7*y+9"):
```



2. Grafik Fungsi Kuadrat

Fungsi kuadrat dua variabel biasanya dinyatakan dalam bentuk

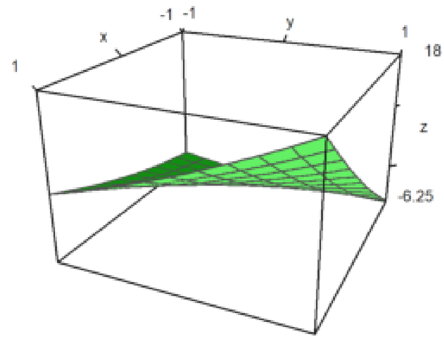
$$f(x,y) = ax^2 + by^2 + cxy + dx + ey + f$$

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:

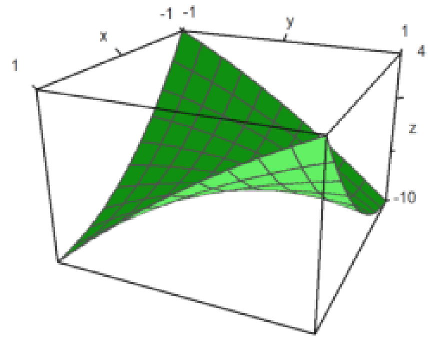
$$f(x, y) = x^2 + y^2 + 4xy + 8x + 3y + 1$$

```
>plot3d("x^2+y^2+4*x*y+8*x+3*y+1"):
```



$$f(x, y) = 3x^2 - y^2 + 7xy - 5$$

```
>plot3d("3*x^2-y^2+7*x*y-5"):
```



3. Grafik Fungsi Akar Kuadrat

Grafik fungsi akar kuadrat dua variabel

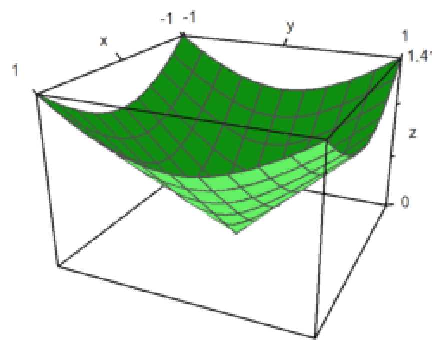
$$f(x, y) = \sqrt{x^2 + y^2}$$

adalah grafik permukaan yang menggambarkan jarak titik (x, y) dari titik asal (0, 0) dalam ruang tiga dimensi.

Contoh:

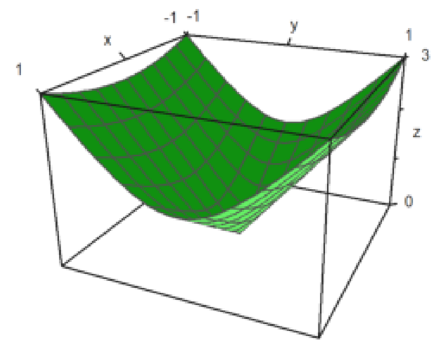
$$f(x,y) = \sqrt{x^2 + y^2}$$

```
>plot3d("sqrt(x^2+y^2)"):
```



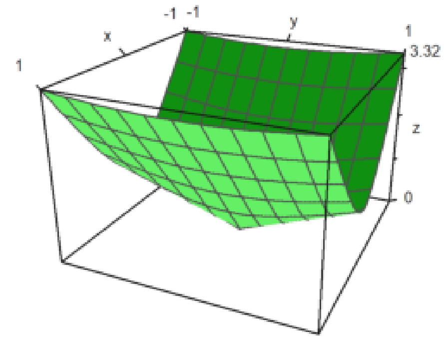
$$f(x,y) = \sqrt{2x^2 + 7y^2}$$

```
>plot3d("sqrt(2*x^2+7*y^2)"):
```



$$f(x,y) = \sqrt{10x^2 + y^2}$$

```
>plot3d("sqrt(10*x^2+y^2)"):
```



4. Grafik Fungsi Eksponensial

Fungsi eksponensial dua variabel bisa dinyatakan

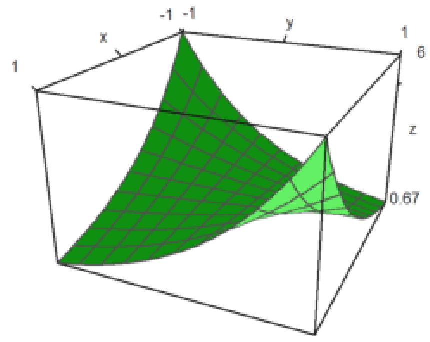
$$f(x, y) = a.b^{xy}$$

dimana a dan b adalah konstanta, x dan y adalah variabel. Fungsi ini menggambarkan pertumbuhan eksponensial yang bergantung pada nilai x dan y.

Contoh:

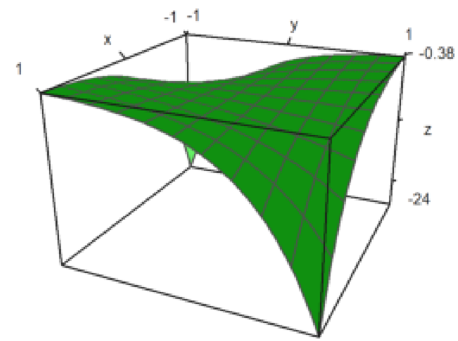
$$f(x,y) = 2.3^{xy}$$

```
>plot3d("2*3^(x*y)"):
```



$$f(x,y) = -3.8^{xy}$$

```
>plot3d("-3*8^(x*y)"):
```



5. Grafik Fungsi Logaritma

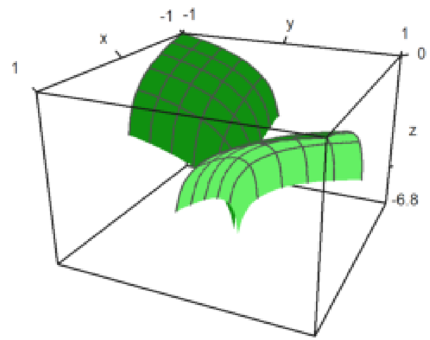
Grafik fungsi logaritma dua variabel adalah grafik yang menggambarkan nilai logaritma dari suatu ekspresi yang melibatkan dua variabel (biasanya x dan y). Fungsi logaritma dua variabel ini dinyatakan sebagai

$$f(x,y) = log_b(xy)$$

dimana b adalah basis logaritma. Basis logaritma ini dapat berbeda-beda.
Contoh:

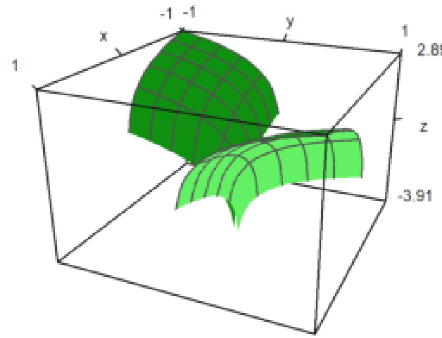
$$f(x, y) = log(xy), basis 10$$

```
>plot3d("log(x*y)"):
```



$$f(x, y) = log(2x.9y), basis 10$$

```
>plot3d("log(2x*9y)"):
```



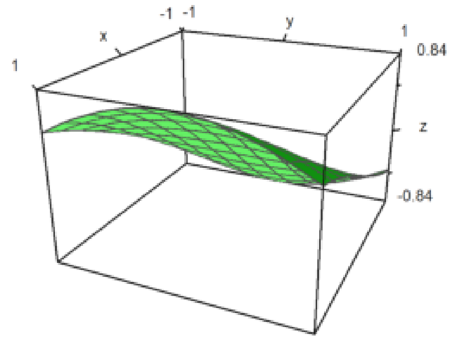
6. Grafik Fungsi Trigonometri

Fungsi trigonometri dua variabel adalah fungsi matematika yang melibatkan operasi trigonometri (seperti sin, cos, tan) pada kedua variabel x dan y.

Contoh:

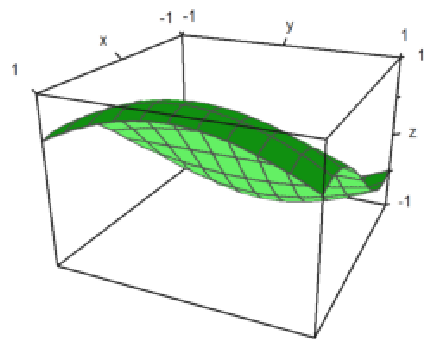
$$f(x, y) = sin(x).cos(y)$$

```
>plot3d("sin(x)*cos(y)"):
```



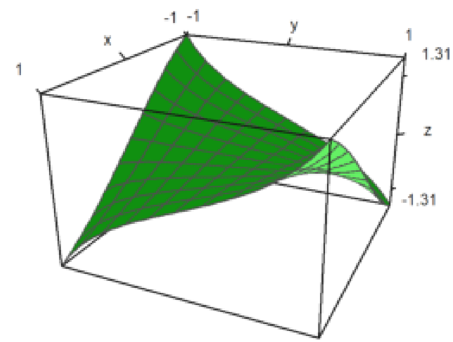
$$f(x,y) = sin(2x).cos(y)$$

```
>plot3d("sin(2x)*cos(y)"):
```



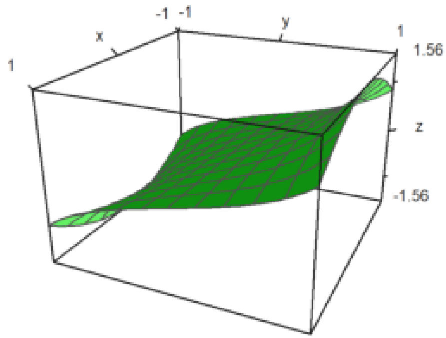
$$f(x, y) = sin(x).tan(y)$$

```
>plot3d("sin(x)*tan(y)"):
```



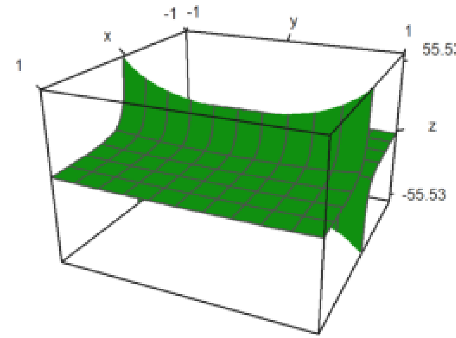
$$f(x,y) = cos(x).tan(y)$$

```
>plot3d("cos(x)*tan(y)"):
```



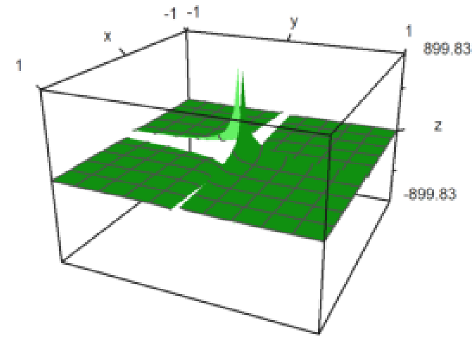
$$f(x,y) = cosec(x).sec(y)$$

```
>plot3d("cosec(x)*sec(y)"):
```



$$f(x, y) = cot(x).cosec(y)$$

```
>plot3d("cot(x)*cosec(y)"):
```



7. Grafik Fungsi Nilai Mutlak

Fungsi nilai mutlak dua variabel, juga dikenal sebagai fungsi modul dua variabel, dinyatakan sebagai

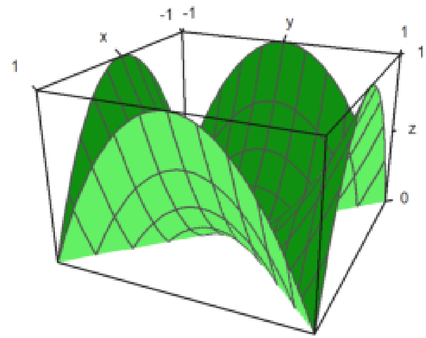
$$f(x, y) = |g(x, y)|$$

dimana g(x,y) adalah fungsi dua variabel.

Contoh:

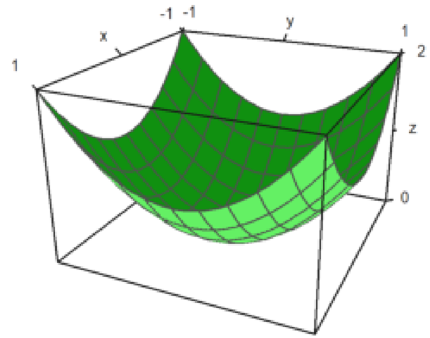
$$f(x, y) = |x^2 - y^2|$$

```
>plot3d("abs(x^2 - y^2)"):
```



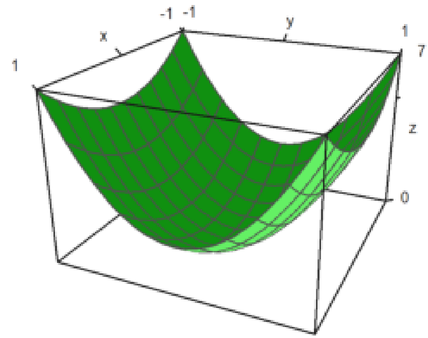
$$f(x, y) = |x^2 + y^2|$$

```
>plot3d("abs(x^2 + y^2)"):
```



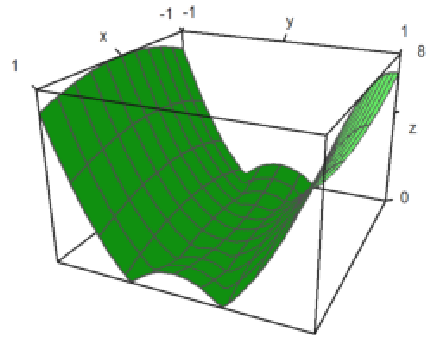
$$f(x, y) = |-2x^2 - 5y^2|$$

```
>plot3d("abs(-2x^2 - 5y^2)"):
```



$$f(x,y) = |x^2 - 8y^2|$$

```
>plot3d("abs(x^2 - 8y^2)"):
```

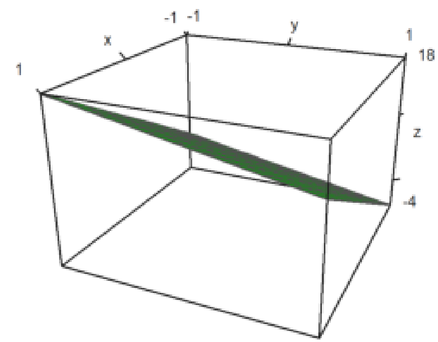


Latihan soal

Buatkan grafik dari fungsi berikut:
1.

$$f(x, y) = 8x - 3y + 7$$

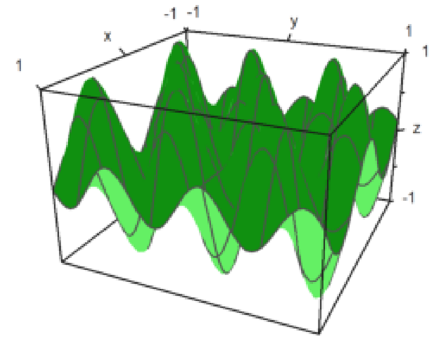
```
>plot3d("8*x - 3*y +7"):
```



2.

$$f(x, y) = cos(5x).sin(9y)$$

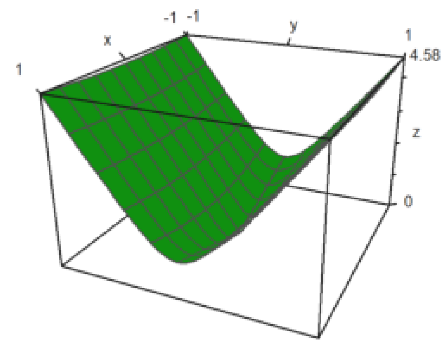
```
>plot3d("cos(5*x)*sin(9*y)"):
```



3.

$$f(x, y) = \sqrt{x^2 + 20y^2}$$

```
>plot3d("sqrt(x^2+20*y^2)"):
```



2) Menggambar Grafik Fungsi Dua Variabel yang Rumusnya Disimpan dalam Variabel Ekspresi

Apa yang dimaksud dengan Grafik Fungsi Dua Variabel?

Grafik fungsi dua variabel adalah representasi visual dari hubungan antara sebuah fungsi matematika dengan dua variabel independen (biasanya disebut sebagai ”x” dan ”y”) dan variabel dependen (biasanya disebut sebagai ”z” atau ”f(x, y)”).

Dalam grafik ini, sumbu x dan sumbu y digunakan untuk menggambarkan nilai-nilai dua variabel independen, sementara permukaan atau grafik 3D digunakan untuk menggambarkan nilai-nilai variabel dependen yang dihasilkan oleh fungsi tersebut.

Grafik fungsi dua variabel membantu memvisualisasikan bagaimana nilai variabel dependen (z) berubah seiring perubahan kedua variabel independen (x dan y) sesuai dengan aturan fungsi tersebut.

Fungsi matematika yg terlibat dalam Menggambar Grafik

Fungsi Dua Variabel

1. Fungsi Linear
Bentuk umum
f(x, y) = ax + by + c, di mana a, b, dan c adalah konstanta. Grafiknya adalah bidang datar.

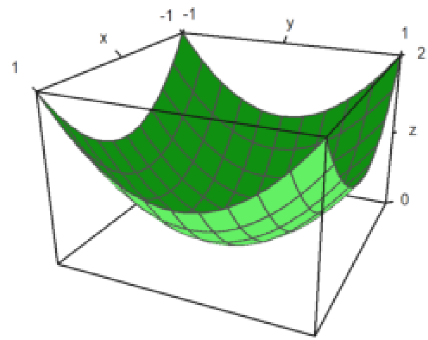
2. Fungsi Kuadratik
Bentuk umum f(x, y) = axˆ2 + byˆ2 + cxy + dx + ey + f.
Grafiknya dapat berupa permukaan yang berbentuk paraboloid, baik terbuka ke atas atau ke bawah.

3.Fungsi Trigonometri
Bentuk umum sinus dan cosinus
(x, y) = sin(x) + cos(y)
akan menghasilkan permukaan yang berulang-ulang naik dan turun.

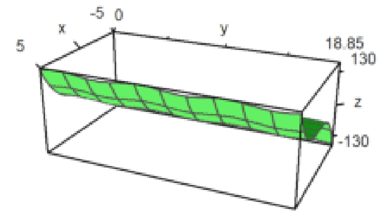
4.Fungsi Pecahan
Bentuk umum f(x, y) = g(x, y) / h(x, y), di mana g(x, y) dan h(x, y) adalah fungsi-fungsi lain.
grafiknya dapat menghasilkan berbagai pola yang tergantung pada sifat fungsi g dan h.

Perintah yang digunakan untuk menggambar grafik fungsi dalam EMT yaitu dengan menggunakan plot3d.
Untuk menampilkan grafik, akhiri perintah plot3d dengan tanda (:).
Tanda (:) akan menampilkan grafik di layar yang berbeda.

```
>plot3d("x^2+y^2"):
```



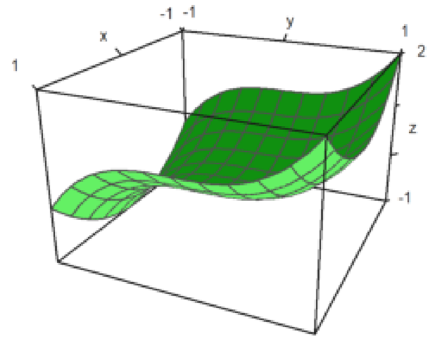

```
>plot3d("x^3+x*sin(y)",-5,5,0,6*pi):
```



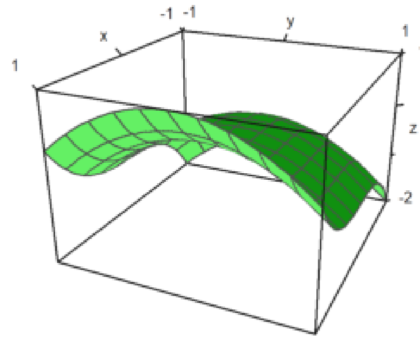
Menyimpan Variabel Ekspresi

Untuk menyimpan sebuah fungsi, dapat dilakukan dengan menggunakan perintah function. Lalu, ketika ingin memanggil atau membuat grafik dari fungsi tersebut, cukup dengan memanggil nama fungsi tersebut.

```
>function a(x,y):= x^2+y^3
>plot3d("a"):
```



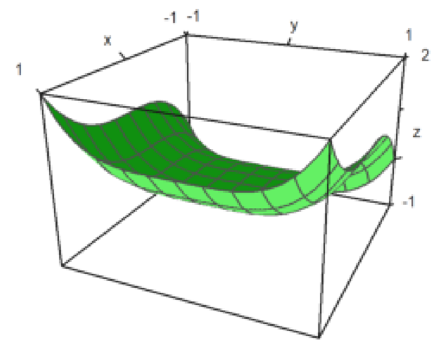

```
>function f(x,y):= x^3-y^2
>plot3d("f"):
```



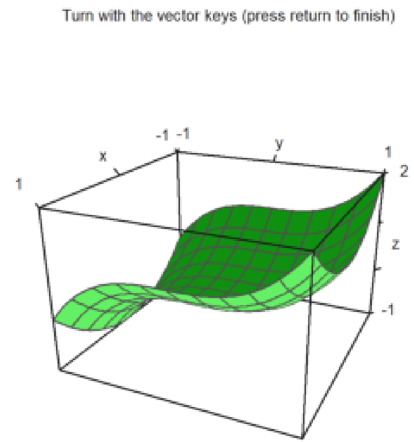
Contoh Latihan Soal

$$f(x,y) = x^3 + y^4$$

```
>function f(x,y):= x^3+y^4
>plot3d("f"):
```

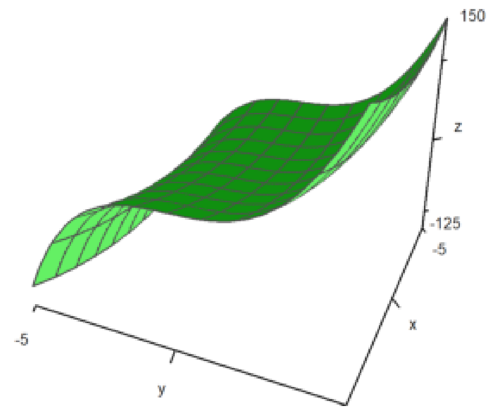



```
>plot3d("a",>user, ...
>title="Turn with the vector keys (press return to finish)"):
```



Perintah ini mengizinkan pengguna untuk menggambar grafik 3D dari fungsi yang mereka masukkan sendiri, serta memberikan petunjuk interaktif tentang cara berinteraksi dengan grafik. Pengguna dapat memutar tampilan grafik menggunakan tombol arah pada keyboard, dan menekan ”return” untuk menyelesaikan interaksi.

```
>plot3d("a",r=5,n=80,fscale=4,scale=1.2,frame=3):
```

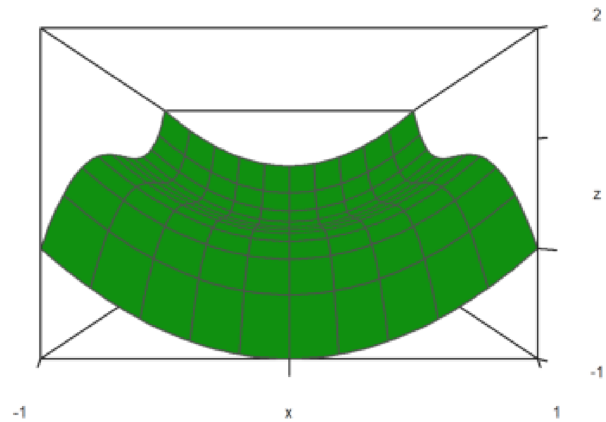


perintah ini akan menggambar grafik tiga dimensi dari fungsi "a" dalam rentang x dan y dari -5 hingga 5, dengan 80 titik untuk detail yang lebih halus. Nilai fungsi akan diperbesar 4 kali, dan grafik akan ditampilkan dengan skala 1.2 untuk tampilan yang lebih jelas. Bingkai grafis akan ditentukan oleh parameter frame=3.

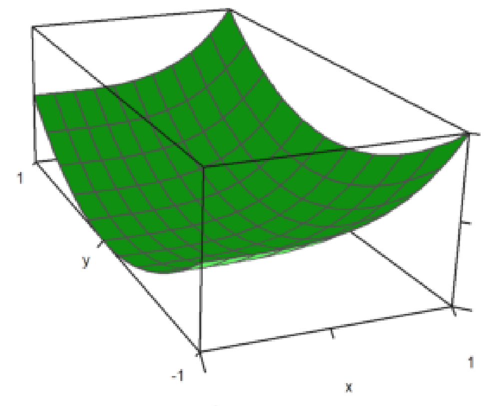
```
>view
```

```
[5,  2.6,  2,  0.4]
```

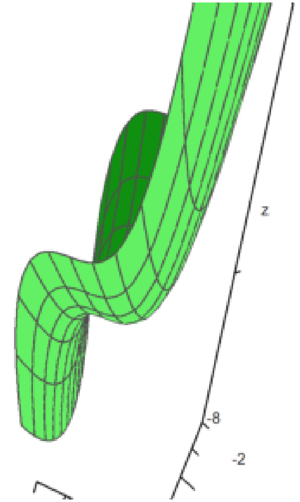
```
>plot3d("a",distance=3,zoom=2,angle=0,height=0):
```



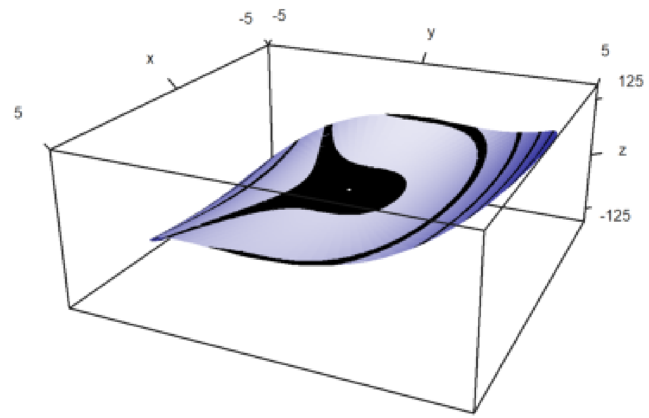

```
>plot3d("x^4+y^2",a=0,b=1,c=-1,d=1,angle=-20°,height=20°, ...
>center=[0.4,0,0],zoom=5):
```



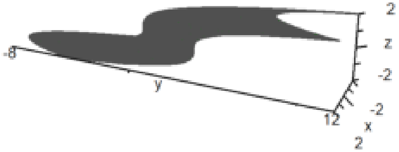

```
>plot3d("a",r=2,<fscale,<scale,distance=13,height=50°, ...
> center=[0,0,-2],frame=3):
```




```
>plot3d("a",r=5,>polar, ...
>fscale=2,>hue,n=100,zoom=4,>contour,color=blue):
```

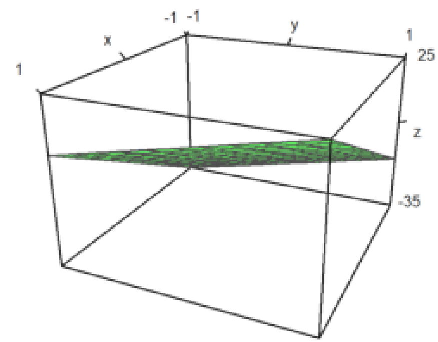



```
>plot3d("x","a","y",r=2,zoom=3.5,frame=3):
```

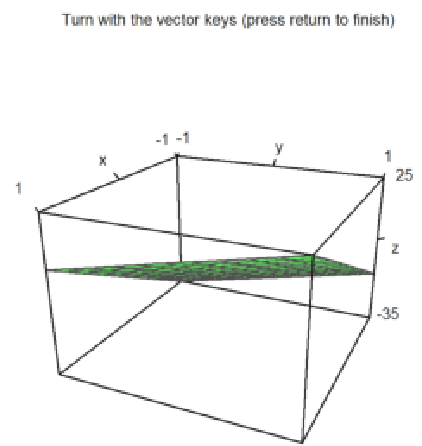


FUNGSI LINEAR

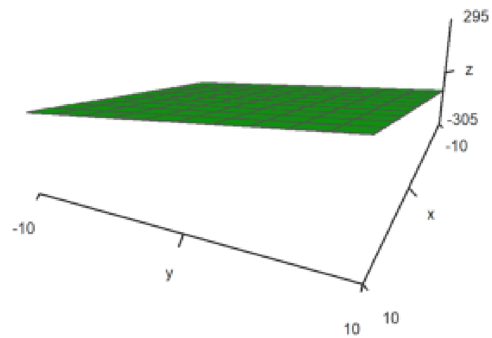
```
>function e(x,y):= 20x+10y-5
>plot3d("e"):
```




```
>plot3d("e",>user, ...
>title="Turn with the vector keys (press return to finish)"):
```



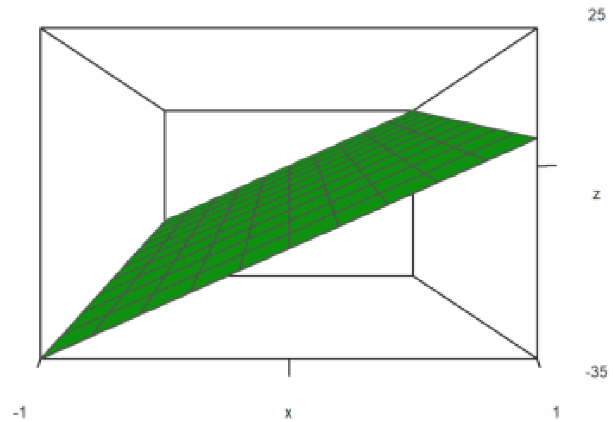

```
>plot3d("e",r=10,n=80,fscale=4,scale=1.2,frame=3):
```



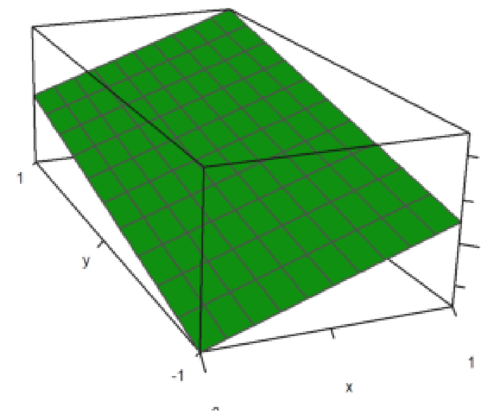

```
>view
```

```
[5,  2.6,  2,  0.4]
```

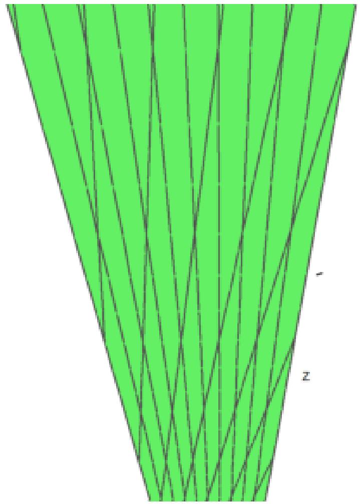
```
>plot3d("e",distance=3,zoom=2,angle=0,height=0):
```



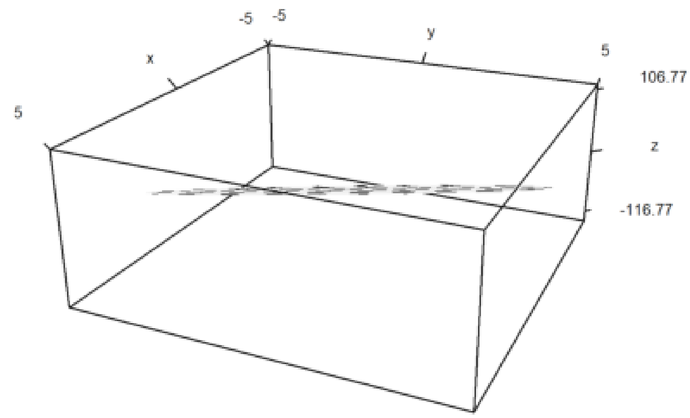

```
>plot3d("e",a=0,b=1,c=-1,d=1,angle=-20°,height=20°, ...
>center=[0.4,0,0],zoom=5):
```




```
>plot3d("e",r=2,<fscale,<scale,distance=13,height=50°, ...
>center=[0,0,-2],frame=3):
```



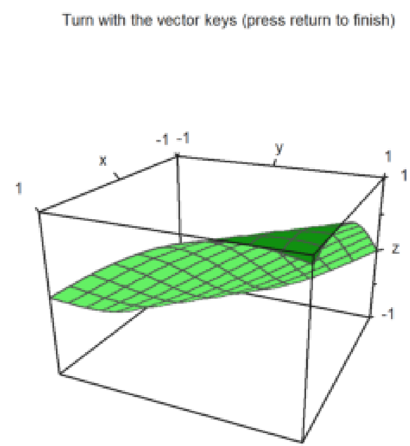
```
>plot3d("e",r=5,>polar, ...
>fscale=2,>hue,n=100,zoom=4,>contour,color=gray):
```



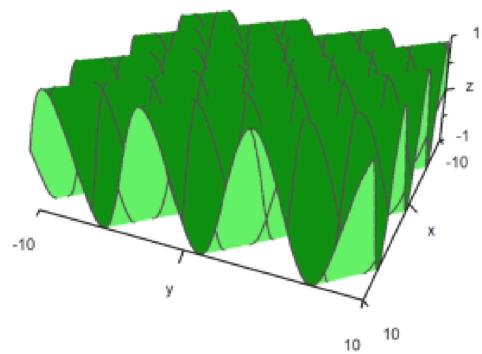

```
>plot3d("x","e","y",r=2,zoom=3.5,frame=3):
```

FUNGSI TRIGONOMETRI

```
>function f(x,y):= sin(x+y)
>plot3d("f")
>plot3d("f",>user, ...
>title="Turn with the vector keys (press return to finish)"):
```



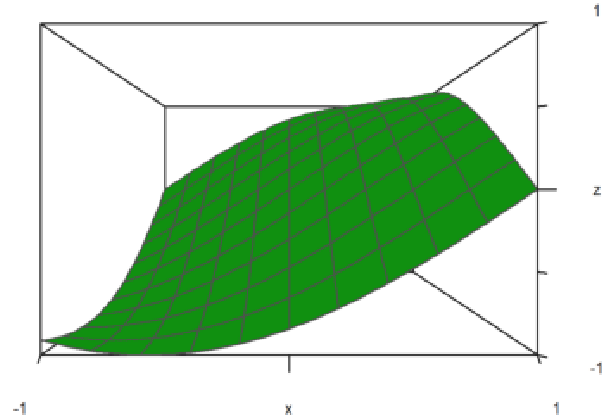

```
>plot3d("f",r=10,n=80,fscale=4,scale=1.2,frame=3):
```



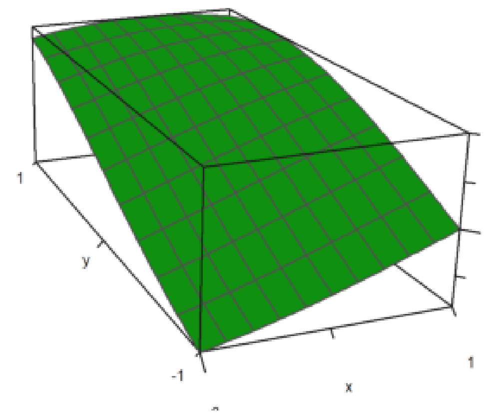

```
>view
```

[5, 2.6, 2, 0.4]

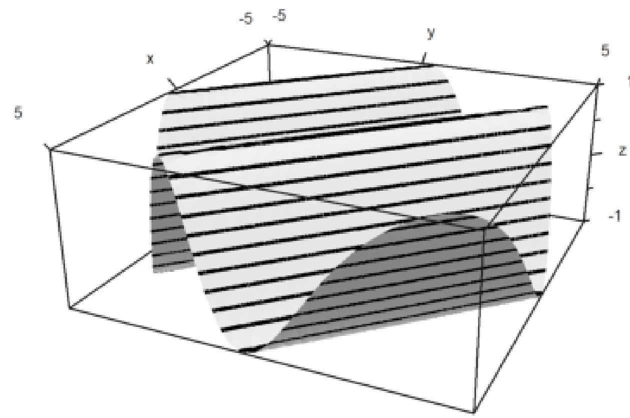
```
>plot3d("f",distance=3,zoom=2,angle=0,height=0):
```




```
>plot3d("f",a=0,b=1,c=-1,d=1,angle=-20°,height=20°, ...
>center=[0.4,0,0],zoom=5):
```




```
>plot3d("f",r=5,>polar, ...
>fscale=2,>hue,n=100,zoom=4,>contour,color=gray):
```



3) Menggambar Grafik Fungsi Dua Variabel yang Fungsinya

Didefinisikan sebagai Fungsi Numerik

Sebelum masuk ke cara memvisualisasikan grafik, perlu diketahui apa itu fungsi dua variabel dan apa itu fungsi numerik.

Fungsi Dua Variabel

Fungsi dua variabel adalah jenis fungsi di mana ada dua variabel bebas (biasanya dinotasikan sebagai x dan y) yang menentukan nilai dari fungsi tersebut. Dengan kata lain, untuk setiap kombinasi nilai dari x dan y, fungsi ini akan menghasilkan satu nilai output tertentu. Fungsi dari dua variabel yang mana setiap kombinasi nilai dari kedua variabel tersebut menghasilkan sebuah nilai tunggal.

Fungsi Numerik

Fungsi dimana setiap pasangan variabel independen adalah angka atau bilangan nyata. Secara sederhana, ketika memasukkan angka atau bilangan nyata ke variabel-variabel dalam fungsi maka hasil akhir yang dihasilkan juga angka atau bilangan nyata, bukan simbol atau ekspresi yang belum dihitung.
Contoh:
ada fungsi

$$f(x, y) = 2x + y$$

Ketika kita memasukkan bilangan nyata ke x dan y maka akan dihasilkan suatu bilangan nyata juga. Misal masukkan x=1 dan y=1. Akan diperoleh

$$2(1) + 1 = 3$$

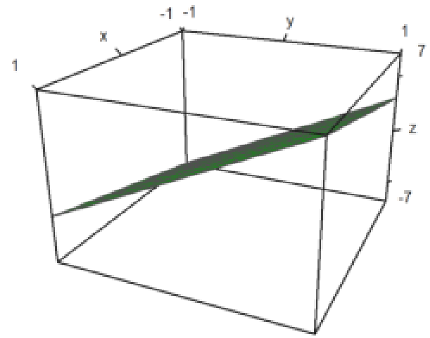
Untuk memvisualisaikan fungsi dua variabel dengan fungsinya didefinisikan sebagai fungsi numerik, akan dibuat grafik 3D dengan sintaks plot3d. Untuk membedakan fungsi numerik dengan simbolik, pada kali ini untuk setiap fungsi numerik dua variabel hanya akan memuat variabel x dan y. Namun, dalam pemakaian secara umum, bisa digunakan variabel apapun.

Penulisan Sintaks:
1) definisikan fungsi numerik
function f(x,y):= ax+by dengan a dan b adalah suatu konstanta dan fungsi tidak selalu direpresentastikan dengan f tetapi bisa dengan huruf apapun. Contoh : g(x,y)
2) sintaks plot3d
plot3d("f"):

Contoh Visualisasi Grafik:
1. Visualisasi grafik fungsi linear dua variabel

$$f(x, y) = 3x + 7y$$

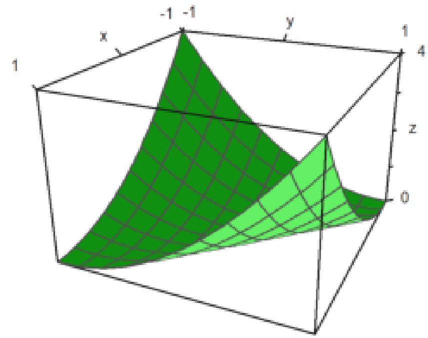
```
>function f(x,y):= 2*x+5*y
>plot3d("f"):
```



2. Visualisasi grafik fungsi kuadrat dua variabel

$$f(x, y) = x^2 + 2xy + y^2$$

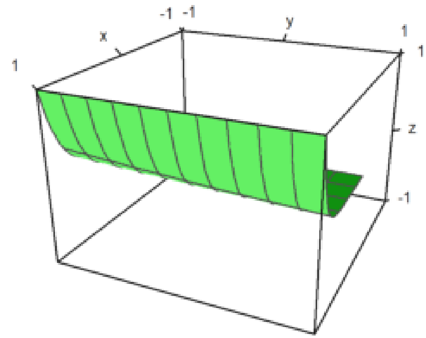
```
>function f(x,y):= x^2+2*x*y+y^2
>plot3d("f"):
```



3. Visualisasi Grafik Fungsi Eksponen Dua Variabel

$$g(x, y) = x^{2y+8}$$

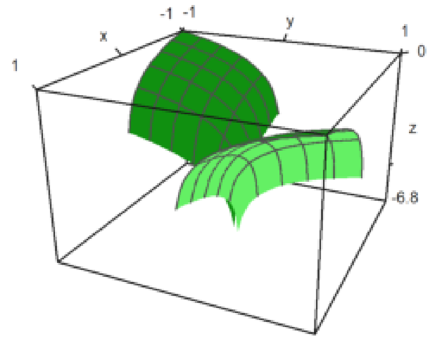
```
>function g(x,y):= x^(2y+8)
>plot3d("g"):
```



4. Grafik Fungsi Logaritma Dua Variabel

$$f(x,y) = \log(xy)$$

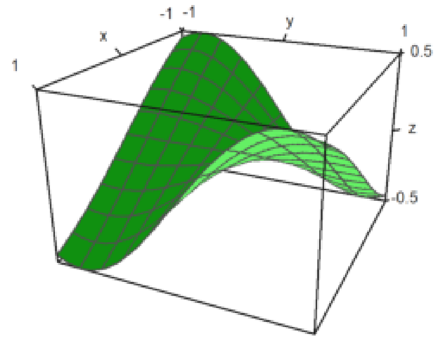
```
>function f(x,y):= log(x*y)
>plot3d("f"):
```



5. Grafik Fungsi Trigonometri Dua Variabel

$$h(x,y) = sin(xy)cos(y)$$

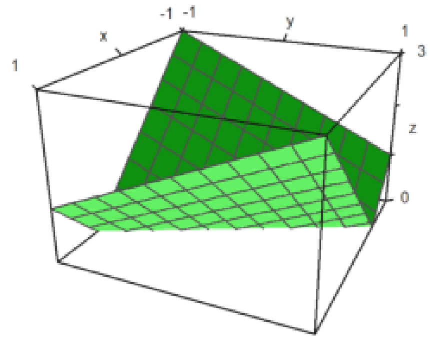
```
>function h(x,y):= sin(x*y)*cos(y)
>plot3d("h"):
```



6. Grafik Fungsi Nilai Mutlak Dua Variabel

$$i(x,y) = |(2x + y)|$$

```
>function i(x,y):= abs(2x+y)
>plot3d("i"):
```

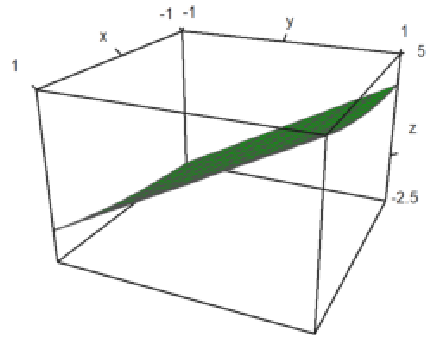


Latihan Soal

Buatlah visualisasi grafik dari fungsi berikut ini!

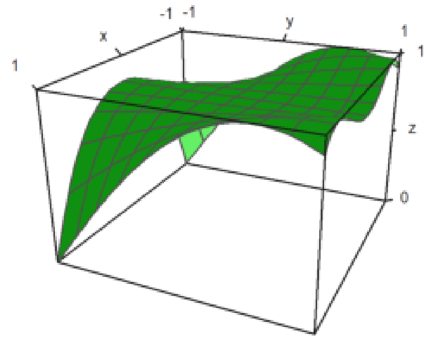
$$f(x, y) = 2^x + 3y$$

```
>function f(x,y):=2^x+3*y
>plot3d ("f"):
```



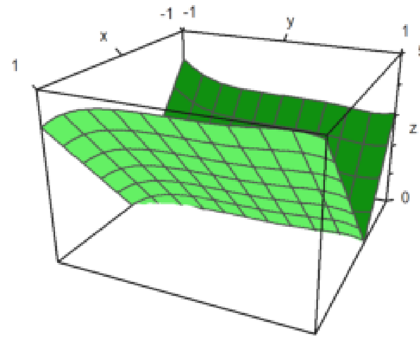
$$g(x,y) = sin(x^2y+1)$$

```
>function g(x,y):= sin(x^2*y+1)
>plot3d("g"):
```



$$h(x, y) = |4x + sin(y^3 + 1)|$$

```
>function h(x,y):=abs(4*x + sin(y^3+1))
>plot3d("h"):
```



4) Menggambar Grafik Fungsi Dua Variabel yang Fungsinya

Didefinisikan sebagai Fungsi Simbolik

Grafik Fungsi dua variabel yang fungsinya didefinisikan sebagai fungsi simbolik adalah suatu grafik yang memvisualisasikan Persamaan Linear Dua Variabel (PLDV) dalam koordinat kartesius dengan fungsinya merupakan fungsi simbolik.

Proses visualisasi ini memungkinkan Kita untuk melihat dan memahami bagaimana fungsi tersebut berperilaku dalam tiga dimensi.

Sedangkan yang dimaksud dengan fungsi simbolik yaitu fungsi yang didefinisikan dalam bentuk matematika simbolik, artinya kita memiliki ekspresi matematika yang menggambarkan hubungan antara dua variabel. misalnya, jika kita memiliki fungsi

$$f(x, y) = x^2 + y^2$$

kita dapat menggambar grafiknya untuk melihat bentuk permukaan yang dihasilkan oleh fungsi ini dalam tiga dimensi. Grafik ini mungkin akan berupa tumpukan parabola yang membentuk kerucut.
Karakteristik fungsi simbolik adalah dengan mengganti tanda := menjadi &=

Perbedaan utama antara fungsi numerik dan fungsi simbolik adalah bahwa fungsi numerik memberikan hasil numerik secara langsung (menghasilkan angka), sementara fungsi simbolik memungkinkan kita untuk bekerja dengan simbol matematika sebelum menghitung nilai numeriknya. Pilihan antara keduanya tergantung pada kebutuhan analisis matematika yang kita lakukan.

Tujuan menggambar grafik fungsi dua variabel adalah untuk memahami pola, sifat, dan hubungan antara dua variabel tersebut secara visual, yang dapat membantu dalam analisis dan pemahaman masalah matematika atau ilmu pengetahuan yang melibatkan fungsi ini.

Langkah-langkah Membuat Grafik

1) Definisikan fungsinya terlebih dahulu. Tentukan fungsi dua variabel yang akan divisualisasikan dalam bentuk simbolik.
Misal diambil fungsi

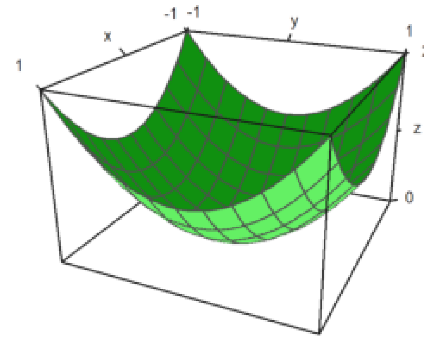
$$f(x,y) = x^2 + y^2$$

Maka perintah yang dapat dituliskan yaitu:

```
>function f(x,y)&= x^2+y^2;
```

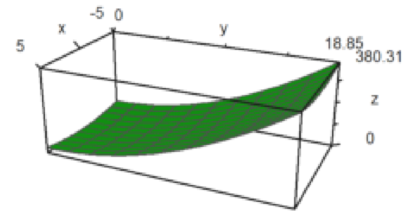
2) Panggil fungsinya

```
>plot3d("f(x,y)"):
```



3) Tentukan rentang variabelnya

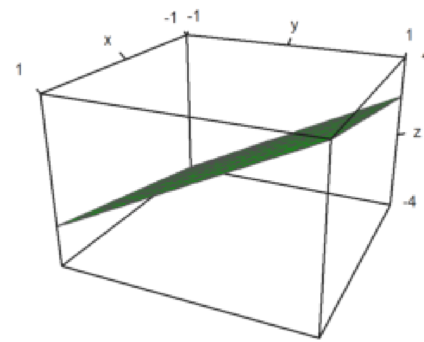
```
>plot3d("f(x,y)",-5,5,0,6*pi):
```



Membuat Grafik Fungsi Linear

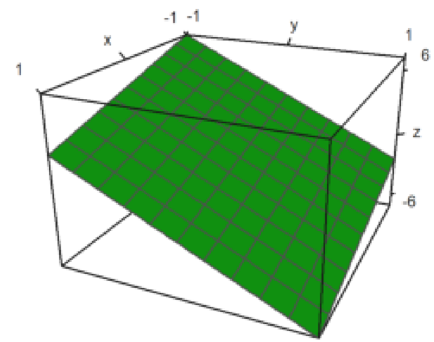
$$g(x,y) = x + 3y$$

```
>function g(x,y) &= x+3*y;
>plot3d("g(x,y)"):
```



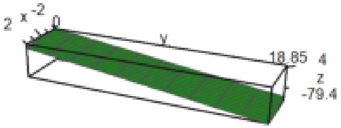
$$M(x,y) = -2x - 4y$$

```
>function M(x,y) &= -2*x-4*y;
>plot3d("M(x,y)"):
```



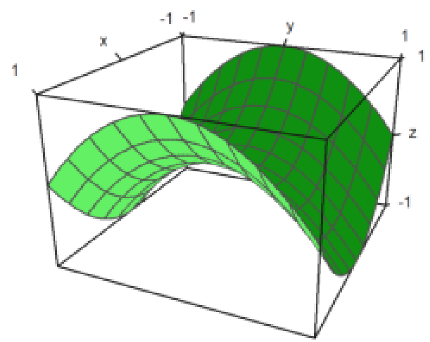
Rentang variabel:

```
>plot3d("M(x,y)",-2,2,0,6*pi):
```


2
x
-2
0
y
18.85
4
z
-79.4

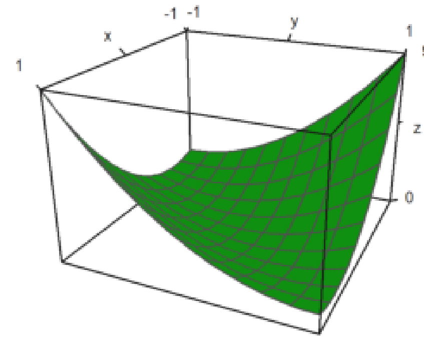
$$Q(x,y) = x^2 - y^2$$

```
>function Q(x,y) &= x^2 - y^2;
>plot3d("Q(x,y)"):
```



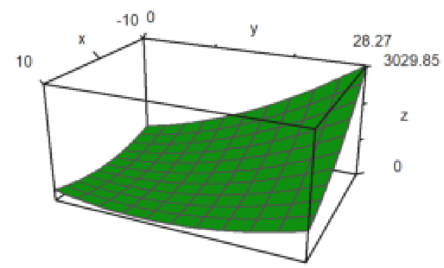
$$P(x,y) = 3x^2 - 4xy + 2y^2$$

```
>function P(x,y) &= 3*x^2-4*x*y+2*y^2;
>plot3d("P(x,y)"):
```



Rentang variabel:

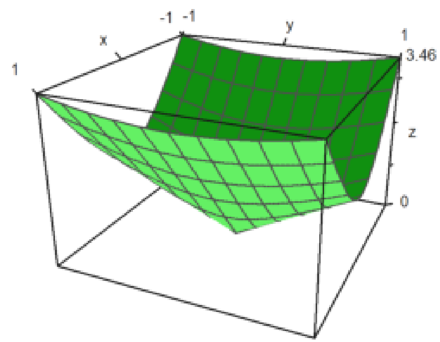
```
>plot3d("P(x,y)",-10,10,0,9*pi):
```



Membuat Grafik Fungsi Akar Kuadrat

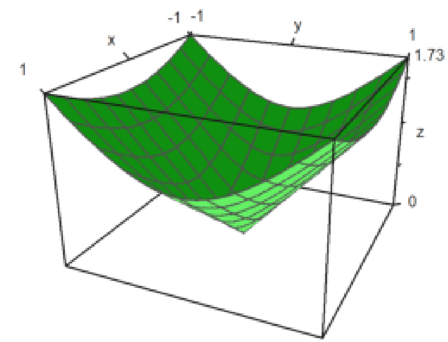
$$A(x, y) = \sqrt{10x^2 + 2y^2}$$

```
>function A(x,y) &= sqrt(10*x^2+2*y^2);
>plot3d("A"):
```



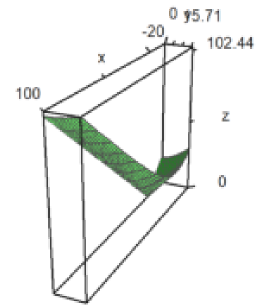
$$W(x,y) = \sqrt{x^2 + 2y^2}$$

```
>function W(x,y) &= sqrt(x^2+2*y^2);
>plot3d("W"):
```



Rentang Variabel:

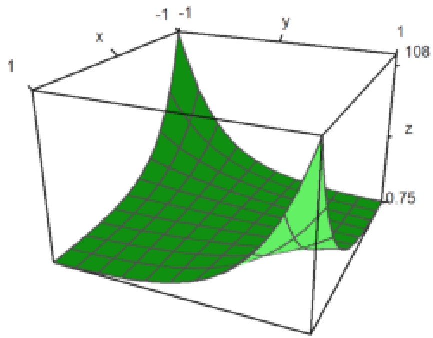
```
>plot3d("W(x,y)",-20,100,0,5*pi):
```



Membuat Grafik Fungsi Eksponensial

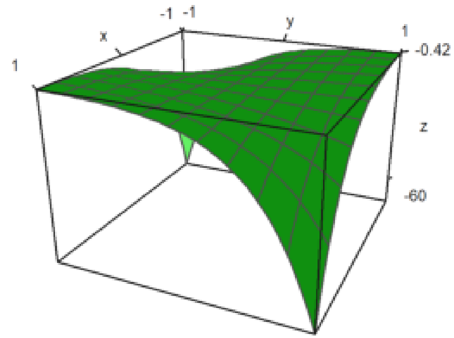
$$F(x, y) = 9.12^{xy}$$

```
>function F(x,y) &= 9*12^(x*y);
>plot3d("F"):
```



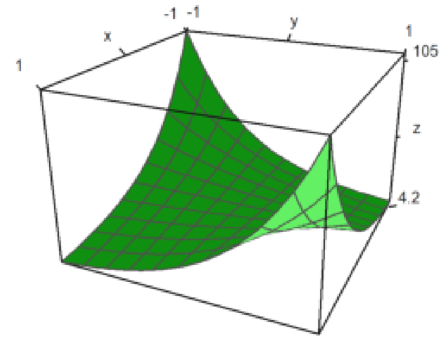
$$H(x, y) = 5.(-20)^{xy}$$

```
>function H(x,y) &= 5*-12^(x*y);
>plot3d("H"):
```



$$T(x,y) = (-21).5^{xy}$$

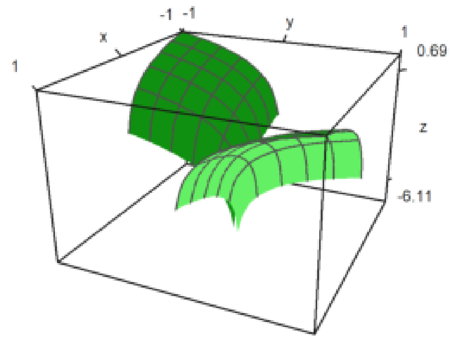
```
>function T(x,y) &= -21*-5^(x*y);
>plot3d("T"):
```



1
x
-1
-1
y
1
105
z
4.2

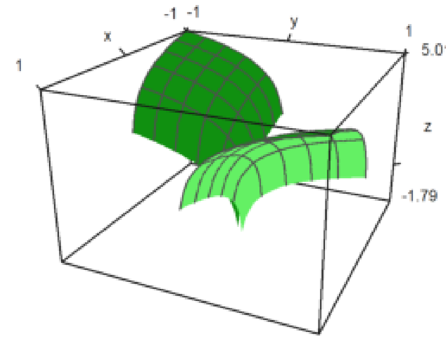
$$B(x, y) = log(x.2y), Basis10$$

```
>function B(x,y) &= log(x*2*y);
>plot3d("B"):
```



$$C(x, y) = log(30x.5y), basis10$$

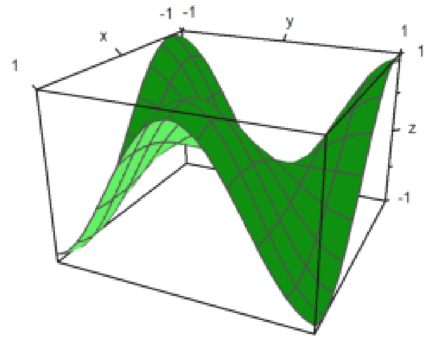
```
>function C(x,y) &= log(30*x*5*y);
>plot3d("C"):
```



Membuat Grafik Fungsi Trigonometri

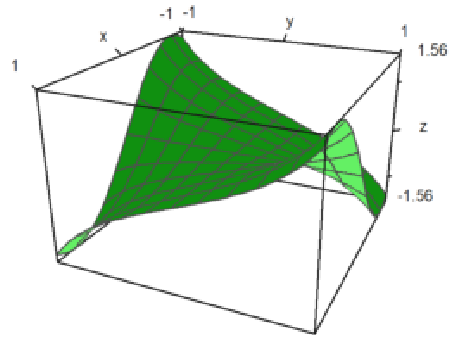
$$D(x,y) = sin(2x).cos(3y)$$

```
>function D(x,y) &= sin(2*x)*cos(3*y);
>plot3d("D"):
```



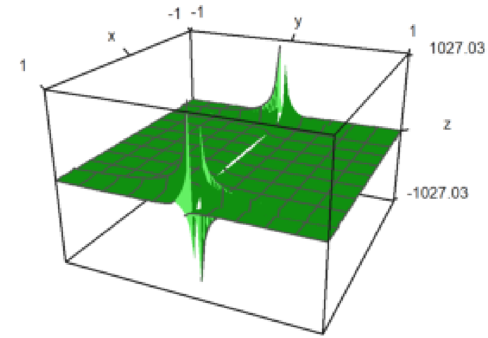
$$J(x,y) = sin(2x).tan(y)$$

```
>function J(x,y) &= sin(2*x)*tan(y);
>plot3d("J"):
```



$$G(x,y) = sec(2x).cot(5y)$$

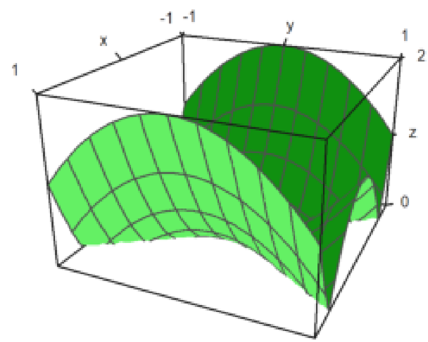
```
>function G(x,y) &= sec(2*x)*cot(y);
>plot3d("G"):
```



Membuat Grafik Fungsi Nilai Mutlak

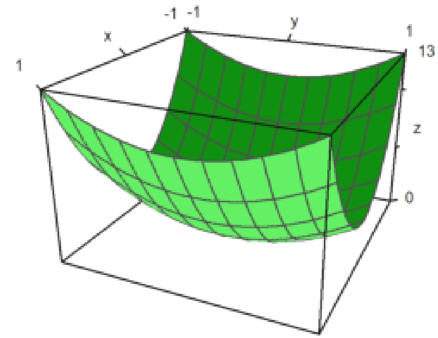
$$T(x,y) = |2x^2 - y^2|$$

```
>function T(x,y) &= abs(2*x^2 - y^2);
>plot3d("T"):
```



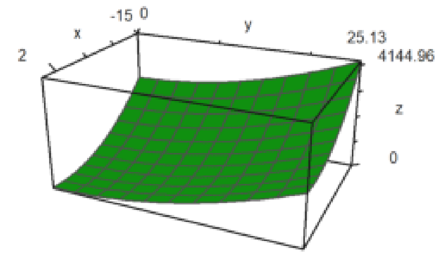
$$M(x,y) = |-10x^2 - 3y^2|$$

```
>function M(x,y) &= abs(-10*x^2 - 3*y^2);
>plot3d("M"):
```



Rentang Variabel :

```
>plot3d("M(x,y)",-15,2,0,8*pi):
```

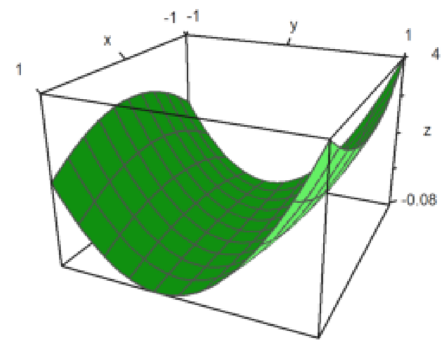


Latihan

Buatlah grafik dari fungsi berikut:

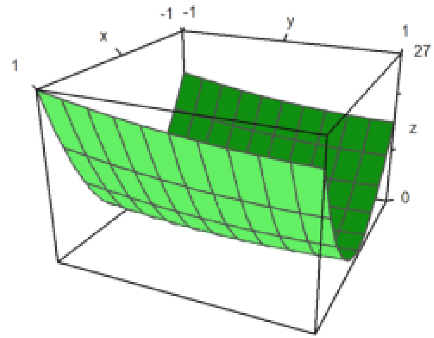
$$A(x, y) = x^2y + 3y^2$$

```
>function A(x,y) &= x^2*y+3*y^2;
>plot3d ("A"):
```



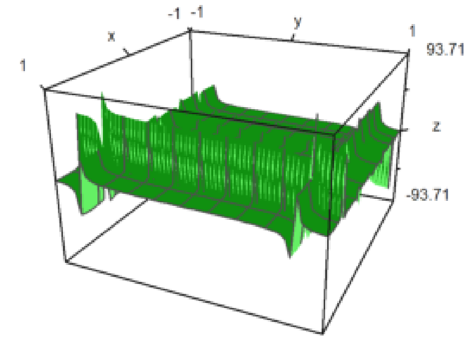
$$B(x, y) = y^2 - 2x^2y + 4x^3 + 20x^2$$

```
>function B(x,y)&= y^2-2*x^2*y+4*x^3+20*x^2;
>plot3d("B"):
```



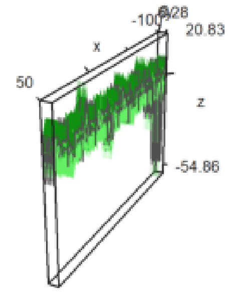
$$C(x,y) = cosec(9x) - tan(2y)$$

```
>function C(x,y)&= cosec(9*x)-tan(2*y);
>plot3d ("C"):
```



Beri rentang variabel untuk fungsi C(x,y):
-100,50,0,2*pi

```
>plot3d("C(x,y)",-100,50,0,2*pi):
```



5) Menggambar Data x, y, z pada ruang Tiga Dimensi (3D)

`Menggambar data pada ruang tiga dimensi (3D) adalah proses`

visualisasi data yang mengubah informasi dalam tiga dimensi, yaitu panjang, lebar, dan tinggi, menjadi representasi visual yang dapat dipahami dan dianalisis.

`Tujuan dari menggambar data 3D adalah untuk membantu pemahaman dan` interpretasi data yang lebih baik, terutama ketika data tersebut memiliki komponen yang tidak dapat direpresentasikan dengan baik dalam dua dimensi.

Sama seperti plot2d, plot3d menerima data. Untuk objek 3D, kita perlu menyediakan matriks nilai x-, y- dan z, atau tiga fungsi atau ekspresi fx(x,y), fy(x,y), fz(x,y).

$$\gamma(t,s) = (x(t,s), y(t,s), z(t,s))$$

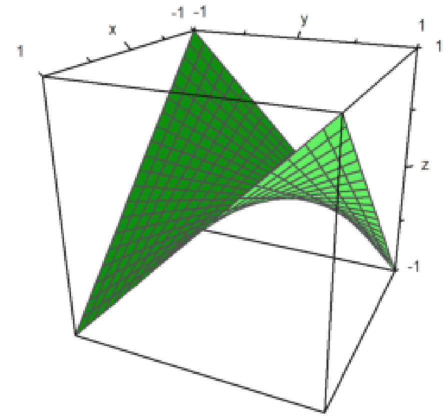
Karena x,y,z adalah matriks, kita asumsikan bahwa (t,s) melalui sebuah kotak persegi. Hasilnya, kita dapat memplot gambar persegi panjang di ruang angkasa.

Kita dapat menggunakan bahasa matriks Euler untuk menghasilkan koordinat secara efektif.

Dalam contoh berikut, kami menggunakan vektor nilai t dan vektor kolom nilai s untuk membuat parameter permukaan bola. Dalam gambar kita dapat menandai daerah, dalam kasus kita daerah kutub.

Contoh 1 grafik fungsi

```
>t=-1:0.1:1; s=(-1:0.1:1)'; plot3d(t,s,t*s):
```



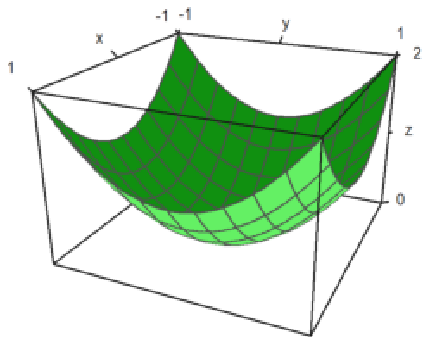
Penjelasan sintaks dari plot
- plot3d = membawa euler untuk mengetahui perintah apa yang harus dilakukan
- (" ...") = tempat kita untuk memasukkan perintah yang kita inginkan

Contoh 2

kita akan memebentuk plot dengan fungsi dibawah ini

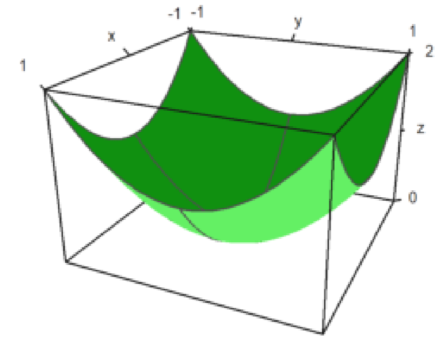
$$x^2 + y^2$$

```
>plot3d("x^2+y^2"):
```



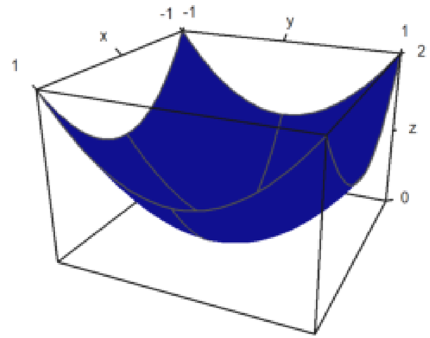
Selanjutnya kita akan menggambar garis pada plot dengan menggunakan grid

```
>plot3d("x^2+y^2",grid=2):
```



Jika kita ingin memodifikasi plot dengan menambahkan warna pada plot, bisa menggunakan fillcolor. Fillcolor dapat diisi dengan 1 warna yang sama atau 2 warna yang berbeda

```
>plot3d("x^2+y^2",grid= 2,fillcolor=[blue,blue]):
```



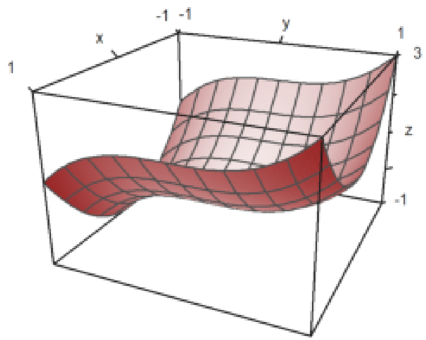
Contoh 3

Jika kita ingin membuat plot 3d pada fungsi

$$2x^2 + y^3$$

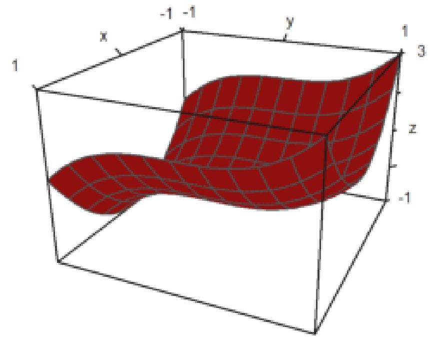
kita bisa menggunakan perintah seperti dibawah ini

```
>plot3d("2x^2+y^3",grid=10,>hue, color=red);
>insimg()
```



Jika kita mau menebalkan warna pada gambar diatas makam bisa menggunakan perintah

```
>plot3d("2x^2+y^3",grid=10,fillcolor=[red,red]);
>insimg()
```

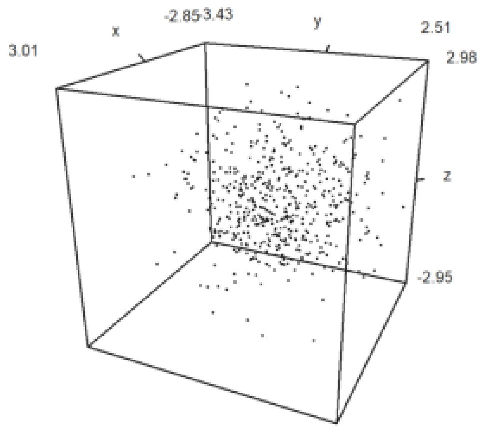


Contoh 4

Tentu saja, titik cloud juga dimungkinkan. Untuk memplot data titik dalam ruang, kita membutuhkan 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="."):
```



Contoh Soal 5

`Dalam contoh berikut, kita membuat tampilan bayangan dari bola yang`

terdistorsi. Koordinat biasa untuk bola adalah

$$\gamma(t,s) = (\cos(t)\cos(s), \sin(t)\sin(s), \cos(s))$$

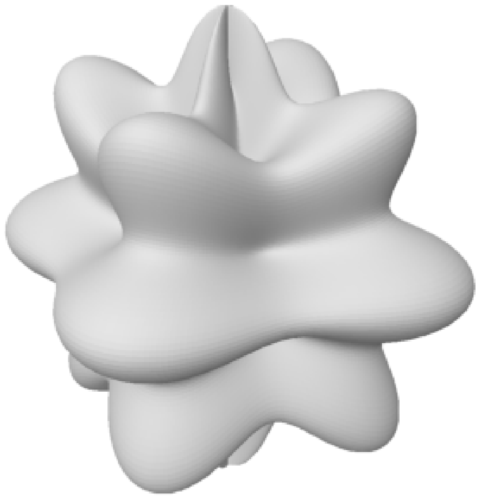
dengan

$$0 \le t \le 2\pi, \quad \frac{-\pi}{2} \le s \le \frac{\pi}{2}.$$

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):
```



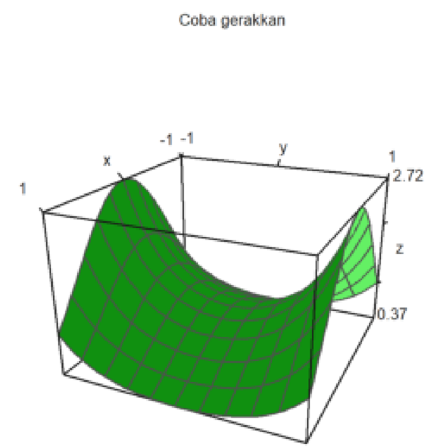
6) Membuat Gambar Grafik Tiga Dimensi (3D) yang Bersifat Interaktif dan animasi grafik 3D

Membuat gambar grafik tiga dimensi (3D) yang bersifat interaktif adalah proses menciptakan visualisasi tiga dimensi yang memungkinkan pengguna berinteraksi dengan objek-objek 3D. Interaktivitas dalam gambar 3D memungkinkan pengguna untuk melakukan tindakan seperti mengubah sudut pandang, memindahkan objek, atau berinteraksi dengan elemen-elemen dalam adegan 3D. Sedangkan animasi grafik 3D dapat mencakup pergerakan, tetapi juga dapat berarti perubahan dalam tampilan atau atribut objek tanpa pergerakan fisik yang mencolok.

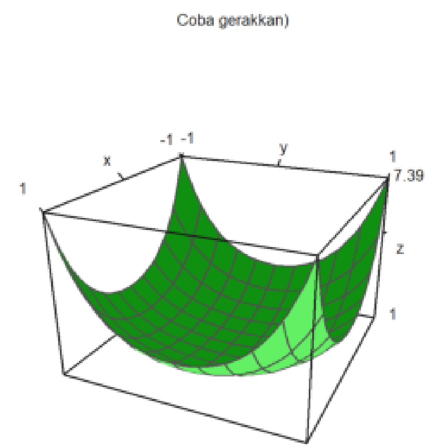
Interaksi user dimungkinkan dengan parameter >user. dengan perintah >user kita dapat menekan tombol berikut.

- kiri, kanan, atas, bawah: memutar sudut pandang
- +,-: memperbesar atau memperkecil
- a: menghasilkan anaglyph (lihat di bawah)
- l : tombol nyalakan sumber cahaya (lihat dibawah)
- spasi: reset ke default
- kembali: akhiri interaksi

```
>plot3d("exp(-x^2+y^2)",>user,...
>title="Coba gerakkan"):
```

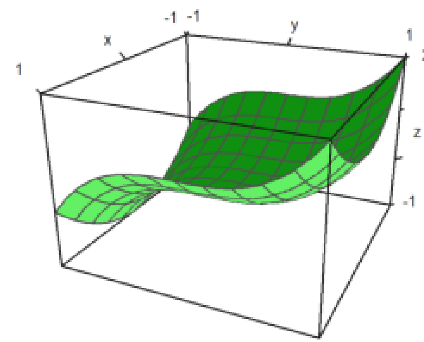



```
>plot3d("exp(x^2+y^2)",>user,...
>title="Coba gerakkan)"):
```



Animasi 3D

```
>function testplot () := plot3d("x^2+y^3");...
>rotate("testplot"); testplot():
```

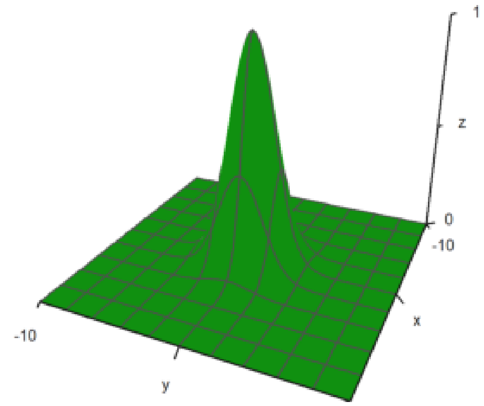


Fungsi rotate yaitu untuk memutar plot.
Fungsi ini akan membuat animasi plot 3D dari fungsi

$$x^2 + y^3$$

yang berputar di sekitar sumbu z dari sudut 0 hingga 360 derajat

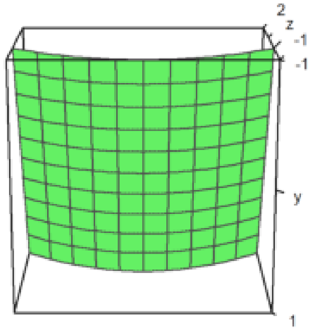
```
>plot3d("exp(-(x^2+y^2)/5)",r=10,n=80,fscale=8,scale=1.2,frame=3,>user):
```



Ada beberapa parameter untuk menskalakan fungsi atau mengubah tampilan grafik.

fscale: menskalakan ke nilai fungsi (defaultnya adalah <fscale).
scale: angka atau vektor 1x2 untuk diskalakan ke arah x dan y.
frame: jenis bingkai (default 1).

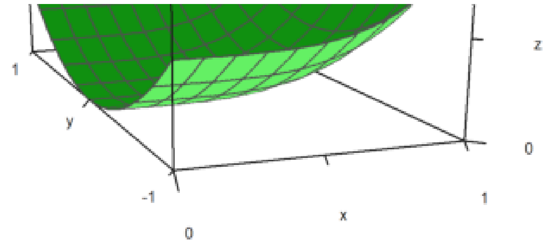
```
>plot3d("x^2+y",distance=10,zoom=5,angle=0,height=5):
```



Tampilan dapat diubah dengan berbagai cara.

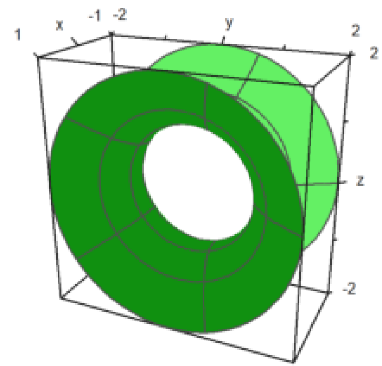
- distance: jarak pandang ke plot.
- zoom: nilai zoom.
- angle: sudut terhadap sumbu y negatif dalam radian.
- height: ketinggian tampilan dalam radian.

```
>plot3d("x^4+y^2",a=0,b=1,c=-1,d=1,angle=-20°,height=20°,...
>center=[0,0,1],zoom=5):
```



Plot selalu terlihat berada di tengah kubus plot. Kita dapat memindahkan bagian tengah dengan center parameter.

```
>plot3d("x^2+1",a=-1,b=1,rotate=true,grid=5):
```



Parameter memutar memutar fungsi dalam x di sekitar sumbu x.

- rotate=1: Menggunakan sumbu x
- rotate=2: Menggunakan sumbu z

7) Menggambar Fungsi Parametrik Tiga Dimensi (3D)

Pengertian

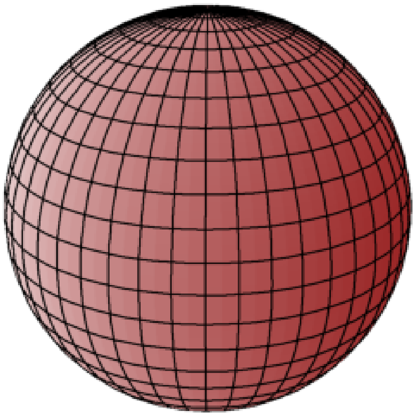
Fungsi parametrik adalah jenis fungsi matematika yang menggambarkan hubungan antara dua atau lebih variabel, di mana setiap variabel dinyatakan sebagai fungsi dari satu atau lebih parameter. Fungsi parametrik digunakan untuk menggambarkan kurva, lintasan, atau hubungan antara berbagai variabel yang bergantung pada parameter-parameter tertentu.

Fungsi parametrik merupakan salah satu cara mendefinisikan kurva atau permukaan dalam ruang 2D atau 3D menggunakan satu atau lebih parameter independen.

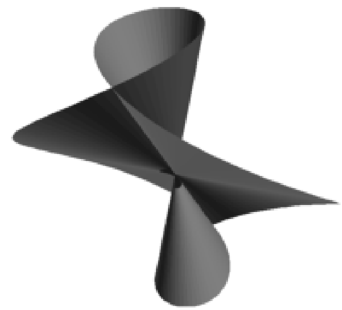
- Dalam 2D, kurva dinyatakan sebagai x(t) dan y(t), di mana adalah t adalah parameter yang mengontrol posisi sepanjang kurva.
- Dalam 3D, kita menggunakan tiga persamaan parametrik untuk mendeskripsikan posisi x,y,z sebagai fungsi dari parameter t.Fungsi ini ditulis sebagai:
x=f(t),
y=g(t),
z=h(t),

Contoh Soal

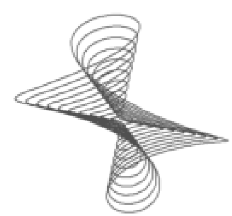
```
>plot3d("cos(x)*cos(y)","sin(x)*cos(y)","sin(y)", a=0,b=2*pi,c=pi/2,d=-pi/2,...
>>hue,color=red,light=[0,1,0],<frame,...
>n=90,grid=[25,50],wirecolor=black,zoom=4):
```



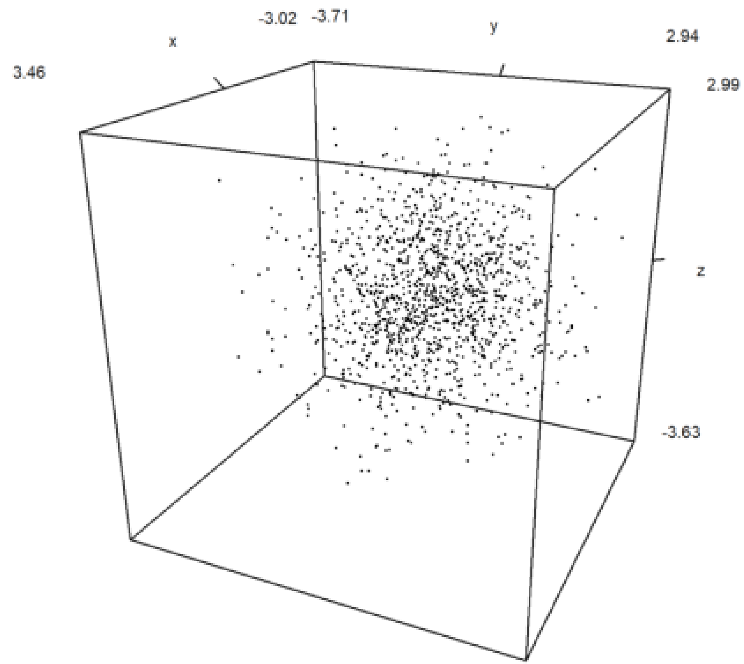
```
>aspect(16/9); allwindow; ...
>x:=linspace(0,2*pi,100); y:=(-1:0.1:1)'; ...
>plot3d(sin(x)*(y/2*sin(x/2)),cos(x)*(y/2*sin(x/2)),y/2*cos(x/2), ...
><frame,hue=2,max=0.5,scale=1.5):
```



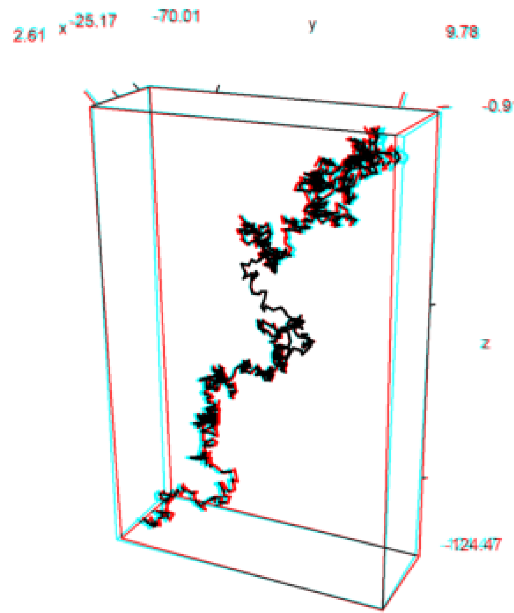
```
>aspect(16/9); allwindow;...
>x:=linspace(0,2*pi,100); y:=(-1:0.1:1)';...
>plot3d(sin(x)*(y/2*sin(x/2)),cos(x)*(y/2*sin(x/2)),y/2*cos(x/2),...
>>lines,<frame,xmin=0,xmax=10,n=10,>user):
```



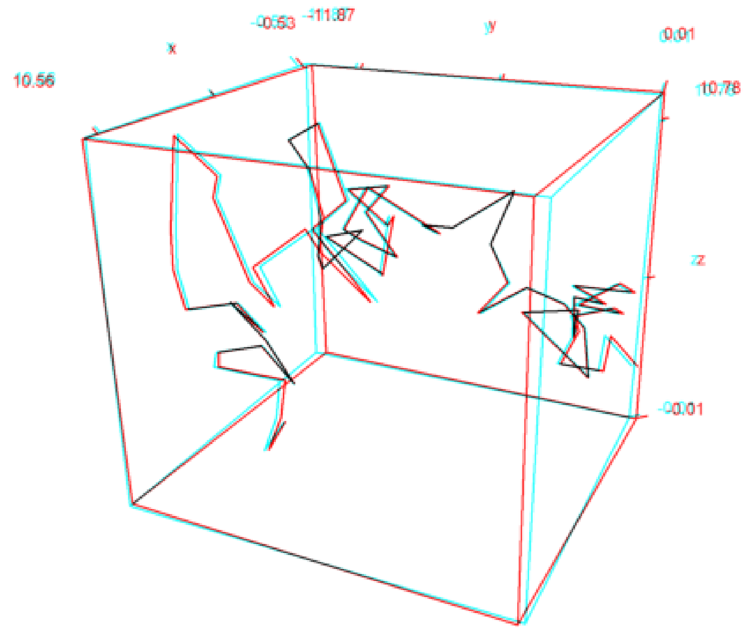
```
>reset;...
>S:=normal(10,1250); plot3d(S[3],S[6],S[9],>points,style="."):
```



```
>S:=normal(10,1250); T:=cumsum(normal(10,1250));...
>plot3d(T[2],T[5],T[8],>wire,...
>linewidth=2,>anaglyph,zoom=3):
```



```
>P=cumsum(normal(5,75));...
>plot3d(P[3],P[4],P[5],>anaglyph,>wire):
```



Fungsi implisit (implicit function) adalah fungsi yang memuat lebih dari satu variabel, berjenis variabel bebas dan variabel terikat yang berada dalam satu ruas sehingga tidak bisa dipisahkan pada ruas yang berbeda.

$$F(x, y, z) = 0$$

(1 persamaan dan 3 variabel), terdiri dari 2 variabel bebas dan 1 terikat

Misalnya, F(x, y, z) = xˆ2 + yˆ2 + zˆ2 - 1 = 0 adalah persamaan implisit yang menggambarkan bola dengan jari-jari 1 dan pusat di (0,0,0).

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.
Solusi dari

$$f(x, y, z) = 0$$

dapat divisualisasikan dalam potongan yang sejajar dengan bidang x-y, x-z, z-x dan y-z.

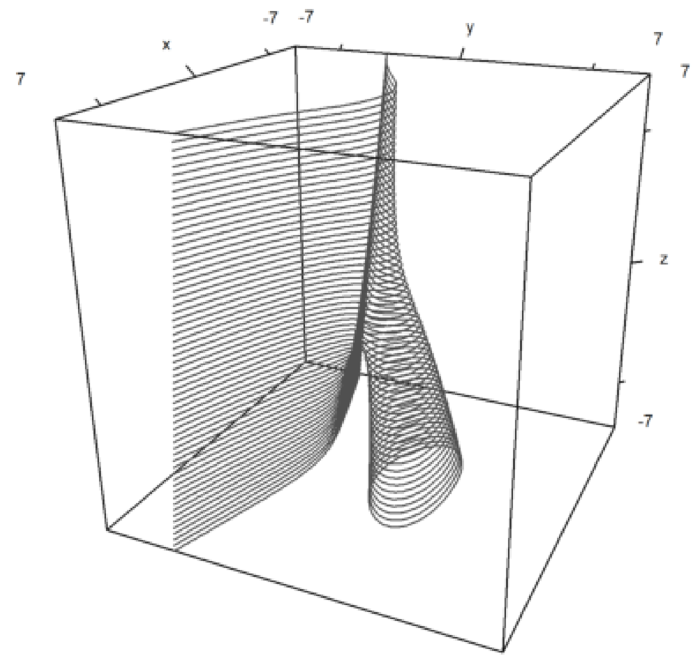
- implisit=1: dipotong sejajar bidang y-z
- implisit=2: dipotong sejajar dengan bidang x-z
- implisit=3: dipotong sejajar dengan bidang z-x (yang berarti pemotongan dilakukan dengan mempertahankan nilai y konstan)
- implisit=4: dipotong sejajar bidang x-y

Tambahkan nilai-nilai ini, jika Anda mau. Dalam contoh kita memplot

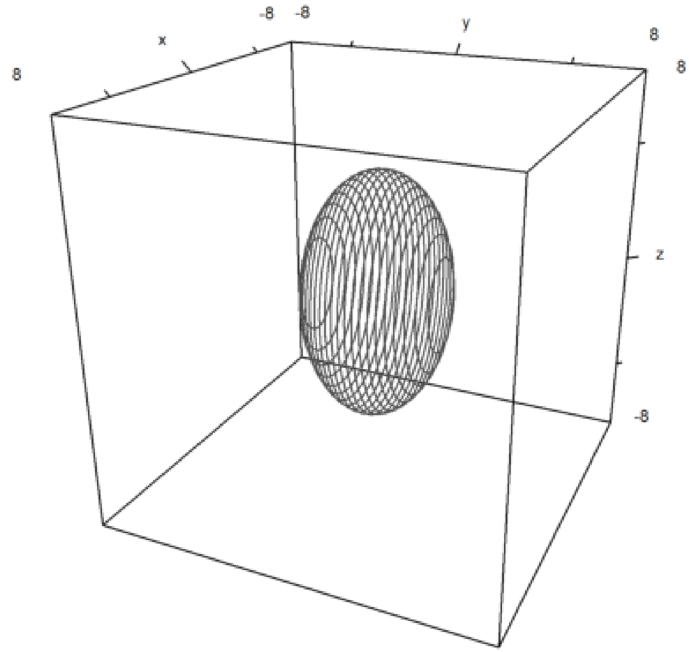
$$M = \{(x, y, z) : x^2 + y^3 + zy = 1\}$$

Contoh Fungsi Implisit

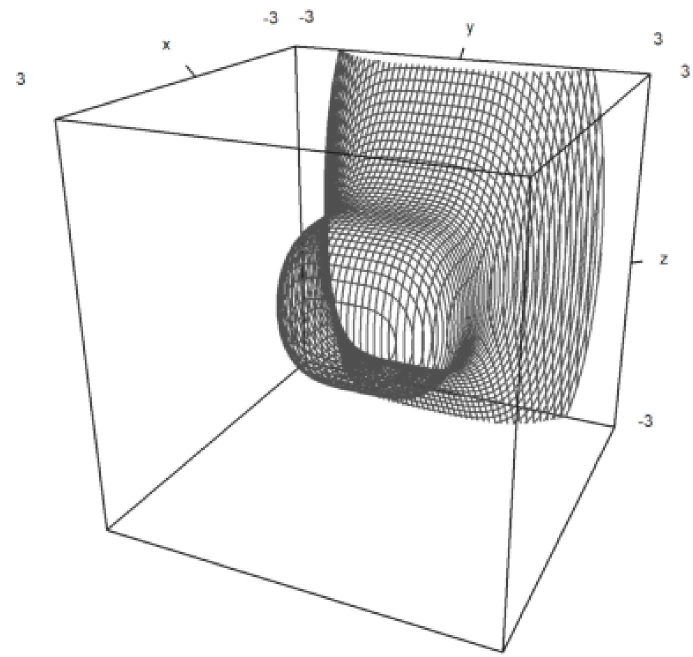
```
>plot3d("x^2+y^3+z*y-1",r=7,implicit=4):
```



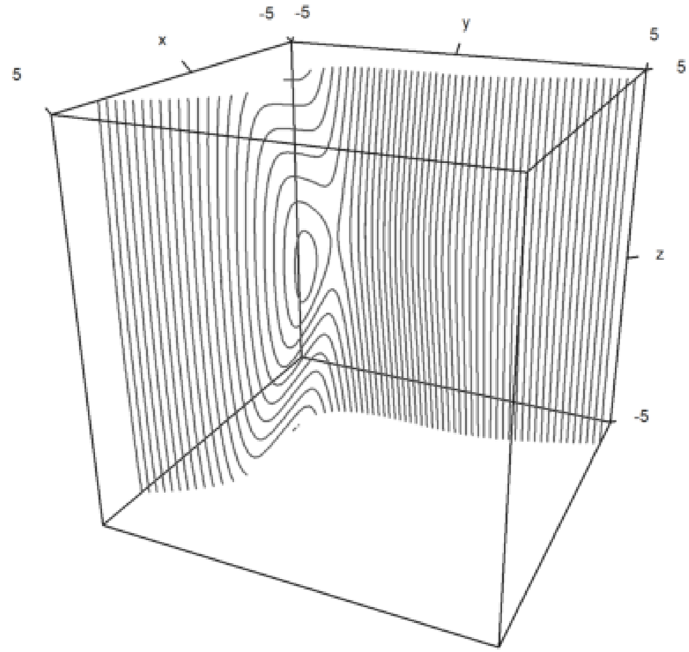

```
>plot3d("2*x^2 + 3*y^2 + z^2 - 25",r=8,implicit=2):
```



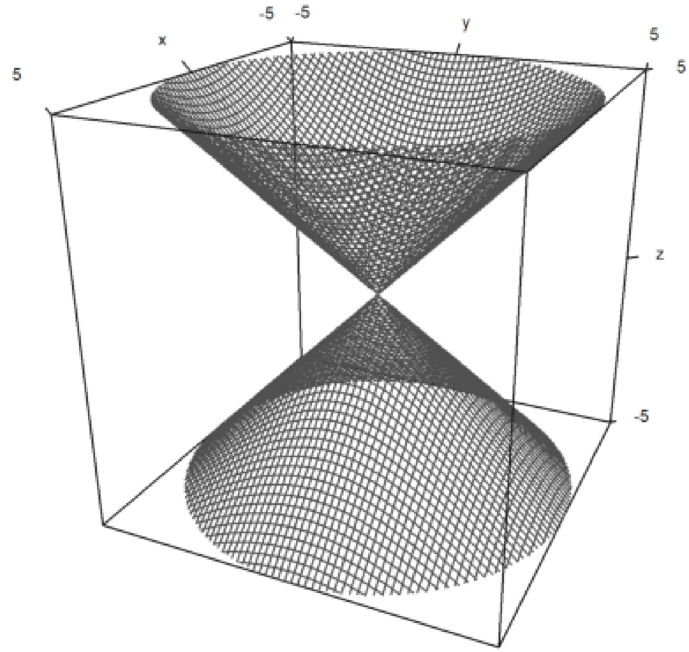

```
>plot3d("4*x^3 + 3*y^4 + 6*z^2 - 10",r=3,implicit=3):
```



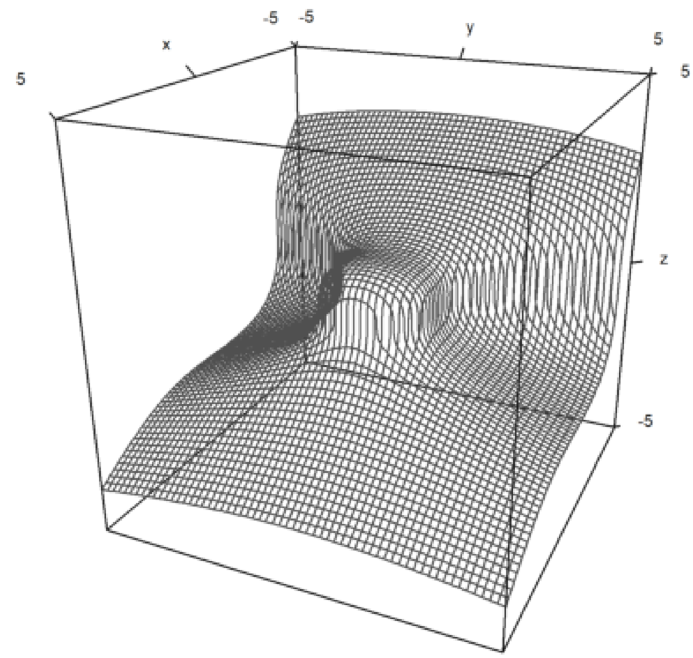

```
>plot3d("x^5 + 5*y^3 + 3*z^2 - 5*x - 7*y - 5*z + 10",r=5,implicit=2):
```



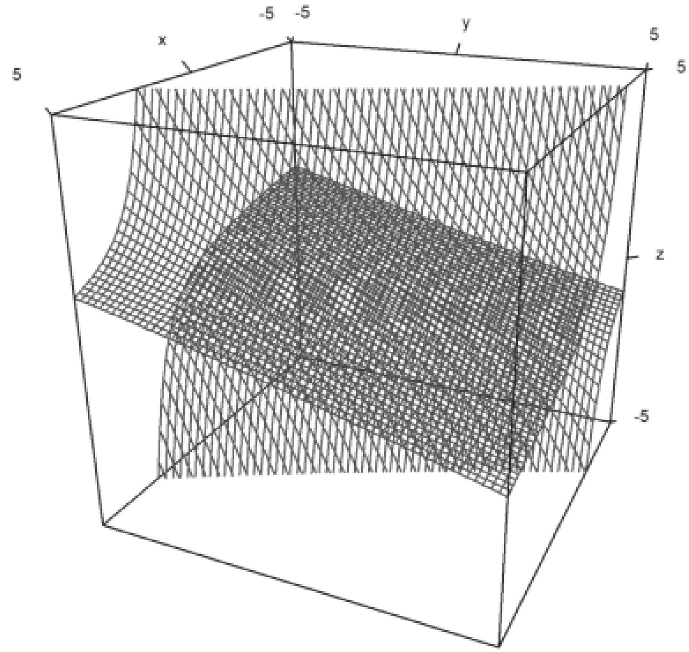

```
>plot3d("x^2 + y^2 - z^2",r=5,implicit=3):
```



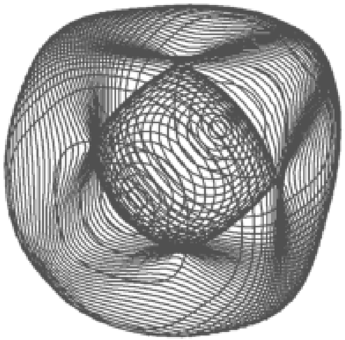

```
>plot3d("x^3 + 2*y^2 +3*z^3-4",r=5,implicit=3):
```



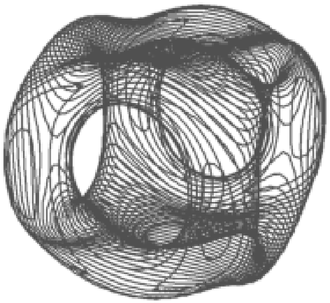

```
>plot3d("x^2+y^2+z^2+2*x*y+4*y*z+8*z*x-20",r=5,implicit=3):
```



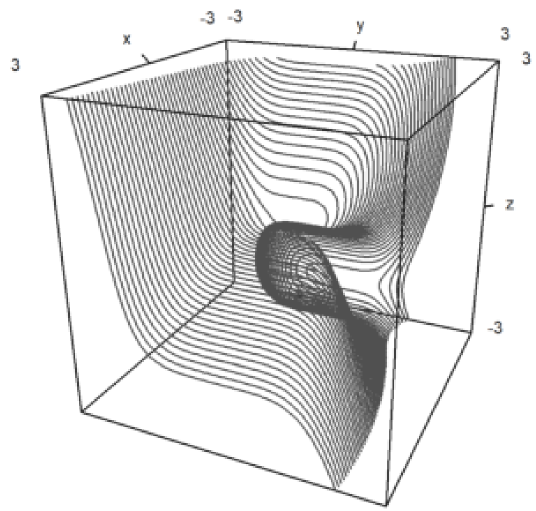

```
>c=1; d=1;
>plot3d("((x^2+y^2-c^2)^2+(z^2-1)^2)*((y^2+z^2-c^2)^2+(x^2-1)^2)*((z^2+x^2-c^2)^2+(y^2-1)^2)-d",r=2,
```



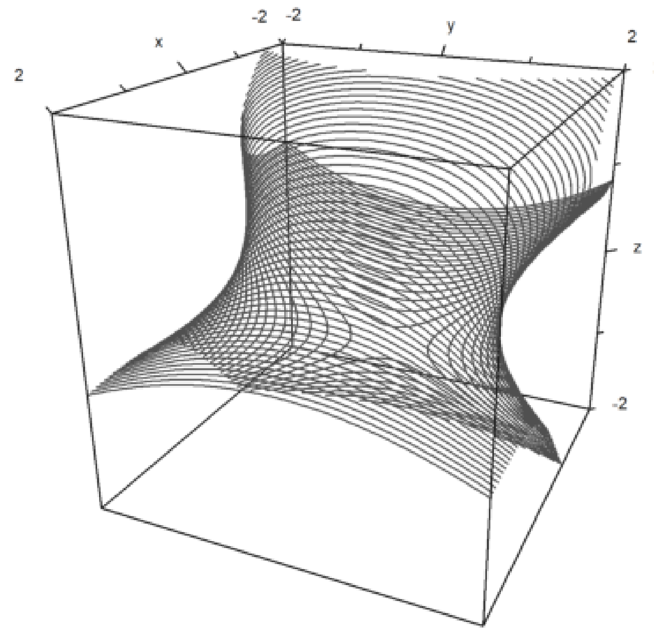
```
>c=1; d=1;
>plot3d("((x^2+y^2+c^2)^2+(z^2-1)^2)*((y^2+z^2-c^2)^2+(x^2-1)^2)*((z^2+x^2-c^2)^2+(y^2-1)^2)-d",r=2,
```



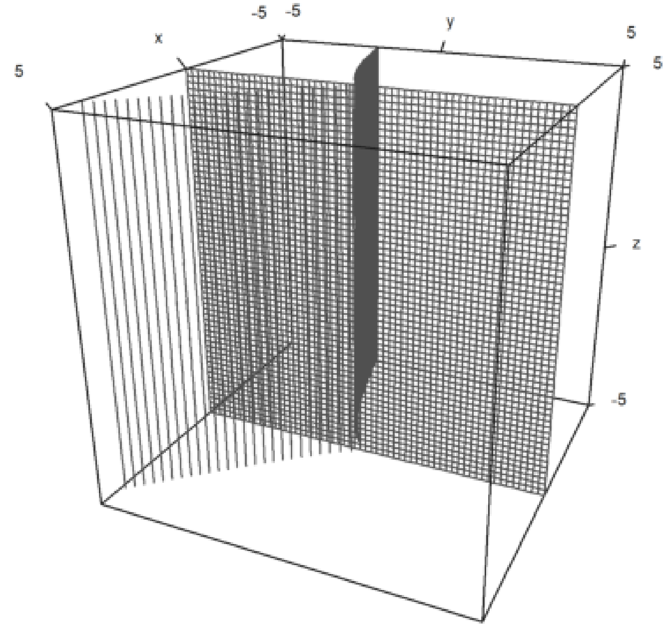
```
>plot3d("x^3+y^5+5*x*z+z^3",>implicit,r=3,zoom=2):
```



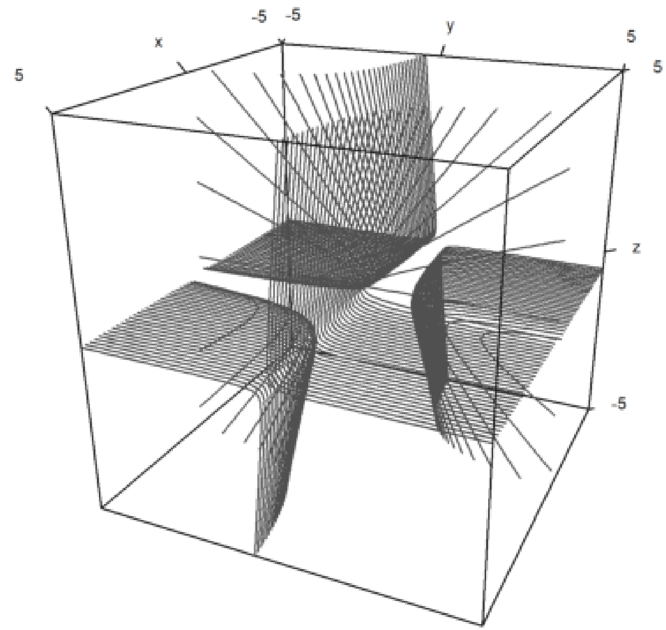

```
>plot3d("x^2+y^2+4*x*z+z^3-2",>implicit,r=2,zoom=2.5):
```




```
>plot3d("x^2*y^2+x^3+y^3*x",>implicit,r=5,zoom=2.5):
```




```
>plot3d("x*y-z^2+2*x*y*z-0",>implicit,r=5,zoom=2.5):
```

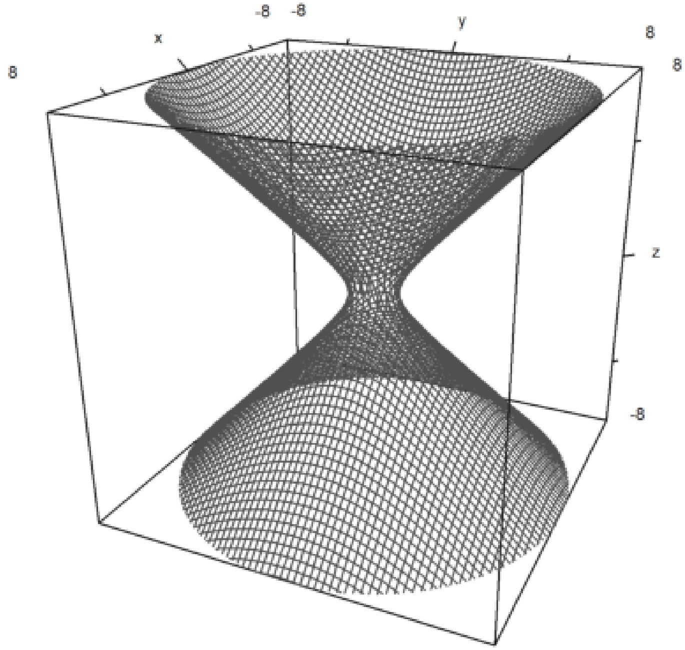


Latihan soal

Gambarlah Fungsi implisit berikut dalam 3D

$$f(x,y,z)=x^2+y^2-z^2-1$$

```
>plot3d("x^2+y^2-z^2-1",r=8,implicit=3):
```

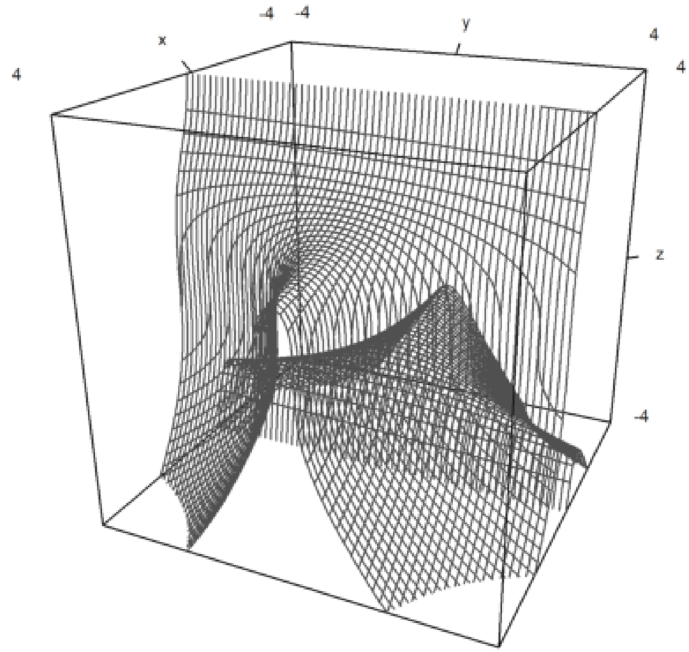


Gambarlah fungsi 3D dari fungsi implisit berikut ini

$$f(x, y, z) = xy + x^3y^2 + xz^3 - 9 = 0$$

dengan r=4

```
>plot3d("x*y+x^3*y^2+x*z^3-9", r=4, implicit=3):
```



9) Mengatur tampilan, warna dan sudut pandang gambar permukaan

permukaan Tiga Dimensi(3D)

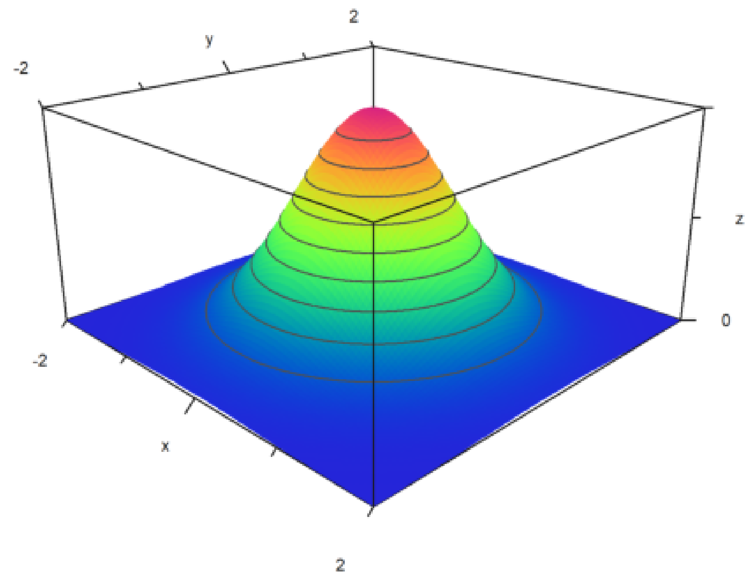
Untuk plot, Euler menambahkan garis grid. Sebagai gantinya dimungkinkan untuk menggunakan garis level dan rona satu warna atau rona berwarna spektral. Euler dapat menggambar tinggi fungsi pada plot dengan bayangan. Di semua plot 3D, Euler dapat menghasilkan anaglyph merah/sian.

-> hue: Menyalakan bayangan cahaya alih-alih kabel.
-> kontur: Memplot garis kontur otomatis pada plot.
- level=... (atau level): Sebuah vektor nilai untuk garis kontur.

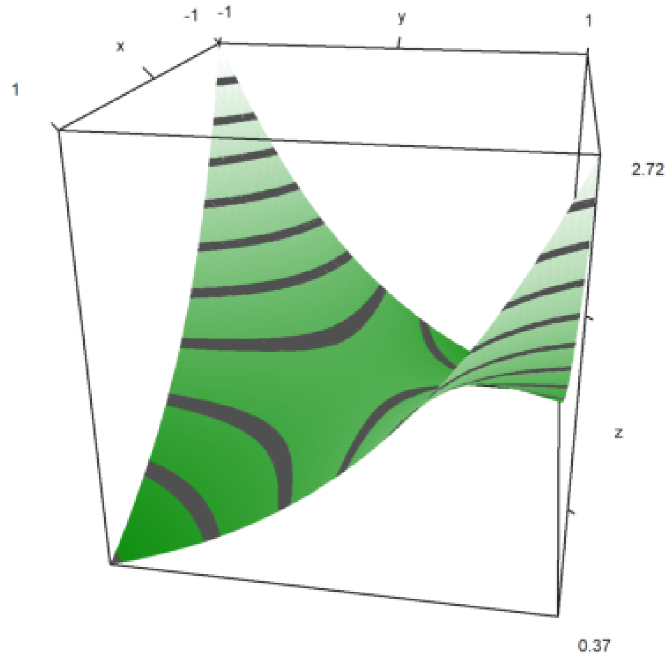
Standarnya adalah level="auto", yang menghitung beberapa garis level secara otomatis. Seperti yang Anda lihat di plot, level sebenarnya adalah rentang level.

Gaya default dapat diubah. Untuk plot kontur berikut, kami menggunakan grid yang lebih halus untuk 100x100 poin, skala 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):
```

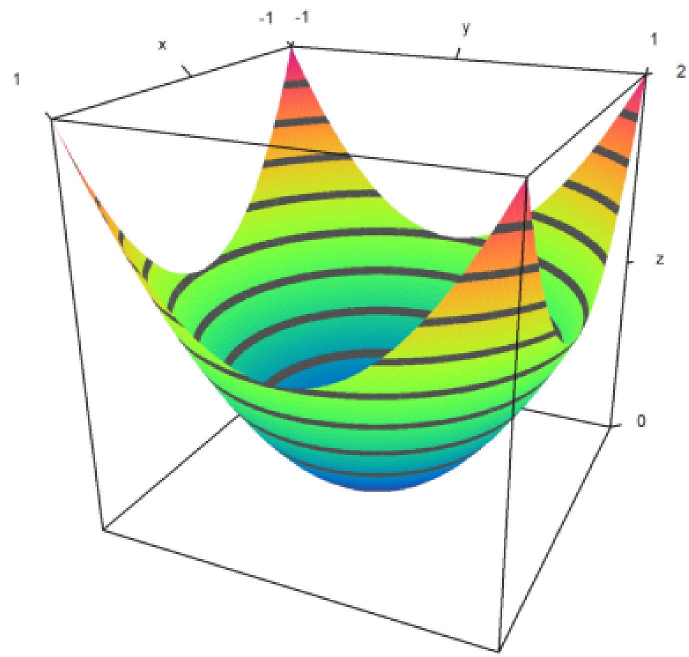


Bayangan default menggunakan warna abu-abu. Tetapi rentang warna spektral juga tersedia.

-> spektral: Menggunakan skema spektral default
- color=...: Menggunakan warna khusus atau skema spektral

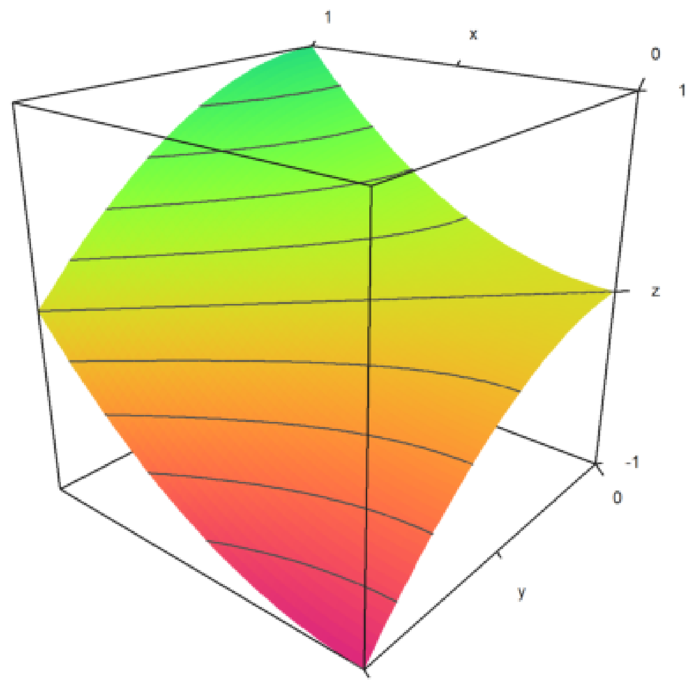
Untuk plot berikut, kami menggunakan skema spektral default dan menambah jumlah titik untuk mendapatkan tampilan yang sangat halus.

```
>plot3d("x^2+y^2",>spectral,>contour,n=100):
```



Alih-alih garis level otomatis, kita juga dapat mengatur nilai garis level. Ini akan menghasilkan garis level tipis alih-alih rentang level.

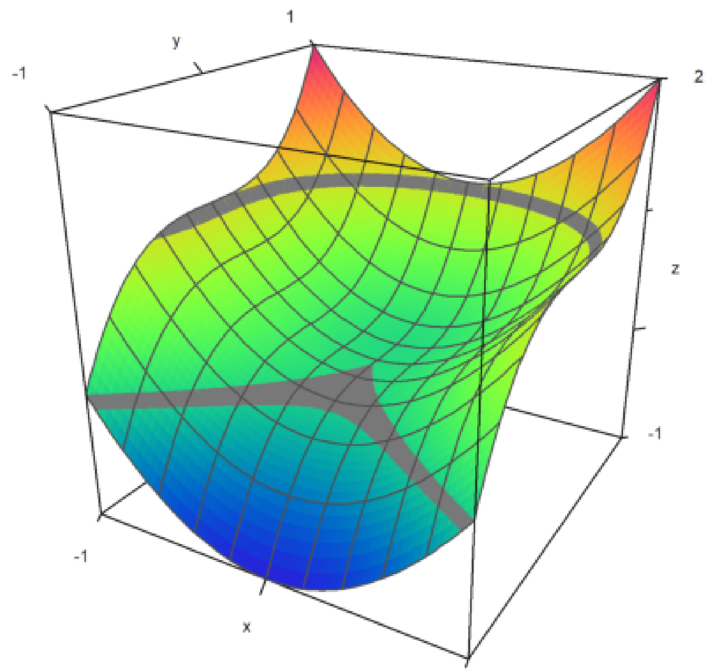
```
>plot3d("x^2-y^2",0,1,0,1,angle=220°,level=-1:0.2:1,color=redgreen):
```



Dalam plot berikut, kami menggunakan dua pita level yang sangat luas dari -0,1 hingga 1, dan dari 0,9 hingga 1. Ini dimasukkan sebagai matriks dengan batas level sebagai kolom.

Selain itu, kami melapisi kisi 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):
```

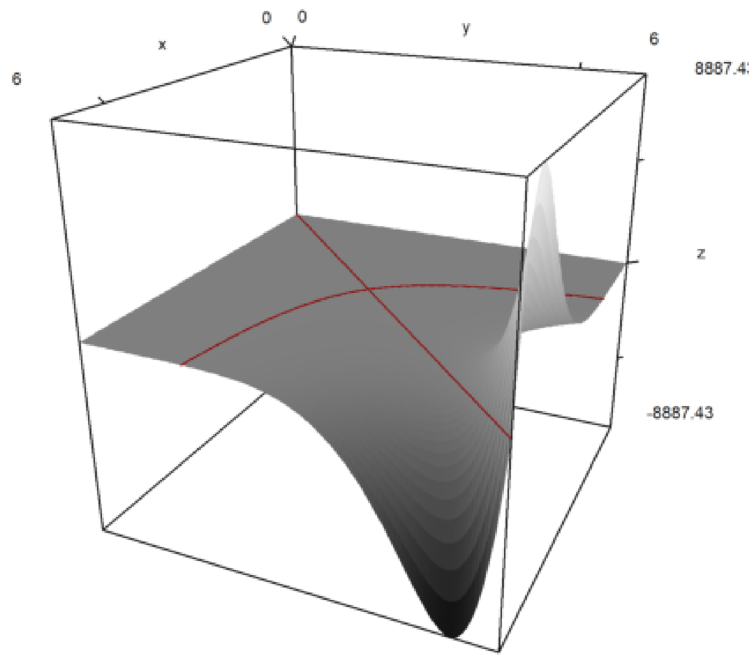


Dalam contoh berikut, kami memplot himpunan, di mana

$$f(x, y) = x^y - y^x = 0$$

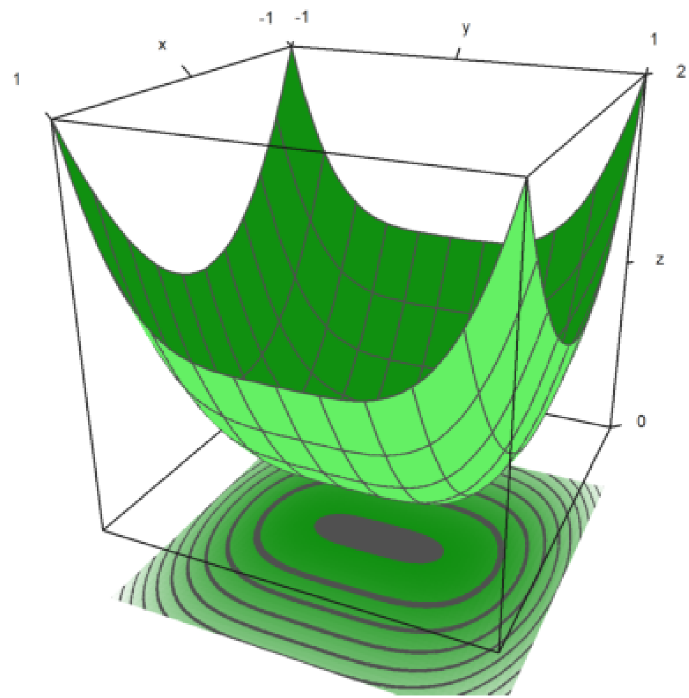
Kami menggunakan satu garis tipis untuk garis level.

```
>plot3d("x^y-y^x",level=0,a=0,b=6,c=0,d=6,contourcolor=red,n=100):
```



Dimungkinkan untuk menunjukkan bidang kontur di bawah plot. Warna dan jarak ke plot dapat ditentukan.

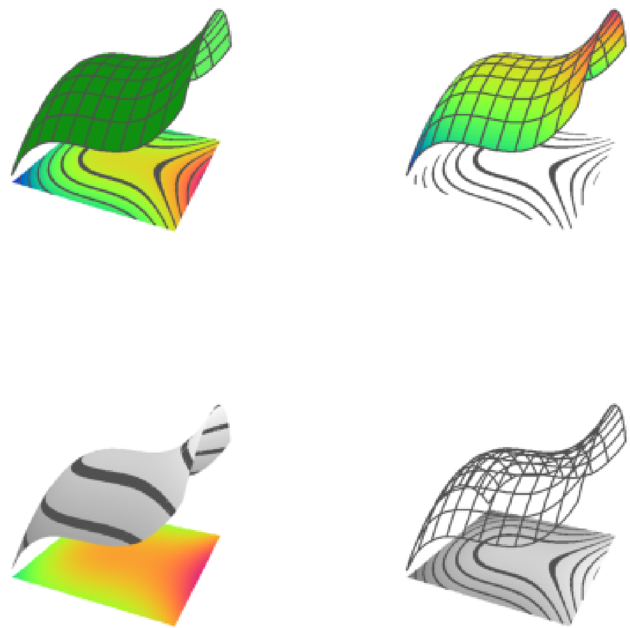
```
>plot3d("x^2+y^4",>cp,cpcolor=green,cpdelta=0.2):
```



Berikut adalah beberapa gaya lagi. Kami selalu mematikan frame, dan menggunakan berbagai skema warna untuk plot dan grid.

```
>figure(2,2); ...
>expr="y^3-x^2"; ...
>figure(1);  ...
```

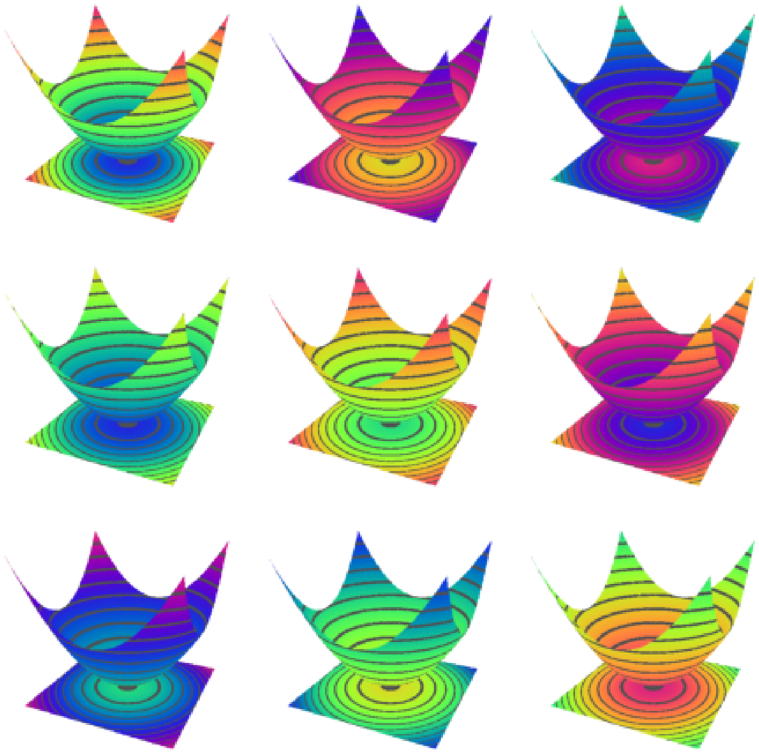
```
>  plot3d(expr,<frame,>cp,cpcolor=spectral); ...
>figure(2);  ...
>  plot3d(expr,<frame,>spectral,grid=10,cp=2); ...
>figure(3);  ...
>  plot3d(expr,<frame,>contour,color=gray,nc=5,cp=3,cpcolor=greenred); ...
>figure(4);  ...
>  plot3d(expr,<frame,>hue,grid=10,>transparent,>cp,cpcolor=gray); ...
>figure(0):
```



Ada beberapa skema spektral lainnya, bernomor dari 1 hingga 9. Tetapi Anda juga dapat menggunakan warna=nilai, di mana nilai

- spektral: untuk rentang dari biru ke merah
- putih: untuk rentang yang lebih redup
- kuningbiru,ungu hijau,birukuning,hijaumerah
- birukuning, hijau ungu, kuning biru, merah hijau

```
>figure(3,3); ...
>for i=1:9;  ...
>  figure(i); plot3d("x^2+y^2",spectral=i,>contour,>cp,<frame,zoom=4);  ...
>end; ...
>figure(0):
```

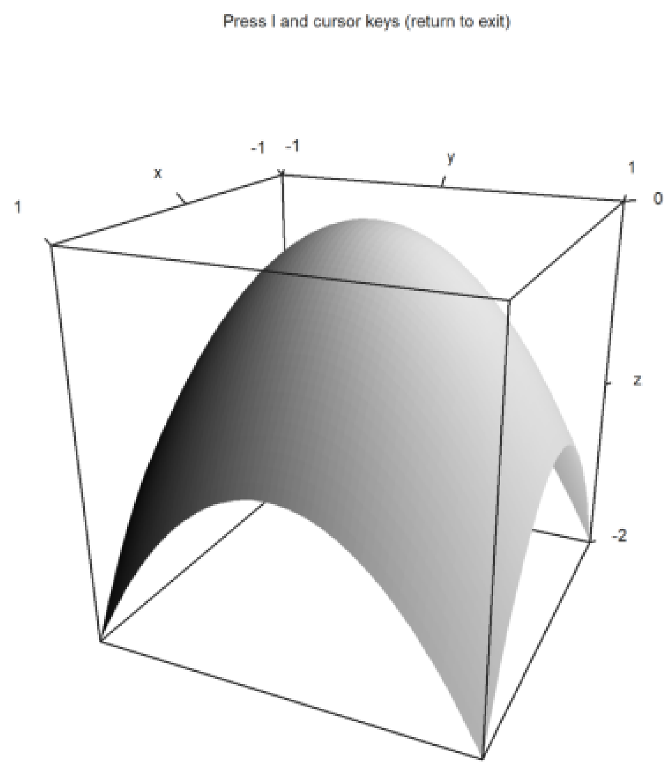


Sumber cahaya dapat diubah dengan l dan tombol kursor selama interaksi pengguna. Itu juga dapat diatur dengan parameter.

- cahaya: arah untuk cahaya
- amb: cahaya sekitar antara 0 dan 1

Perhatikan bahwa program tidak membuat perbedaan antara sisi plot. Tidak ada bayangan. Untuk ini, Anda perlu 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)"):
```

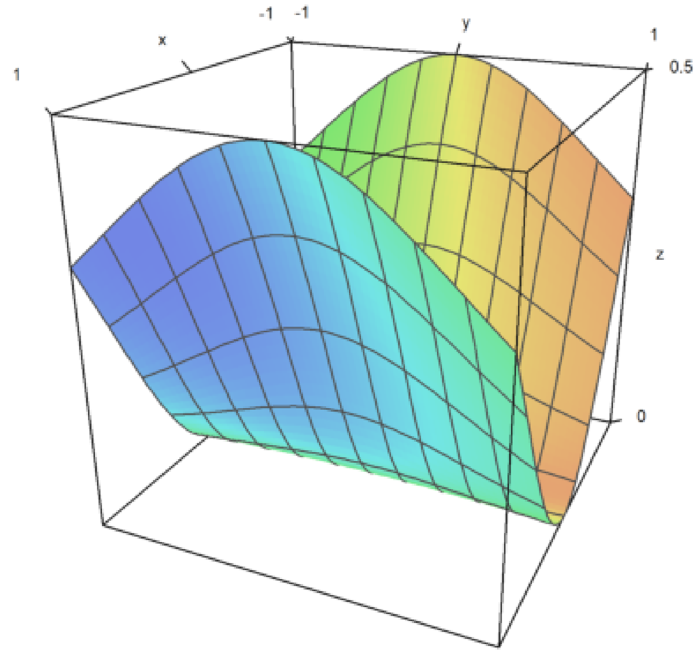


Parameter warna mengubah warna permukaan. Warna garis level juga dapat diubah.

```
>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):
```

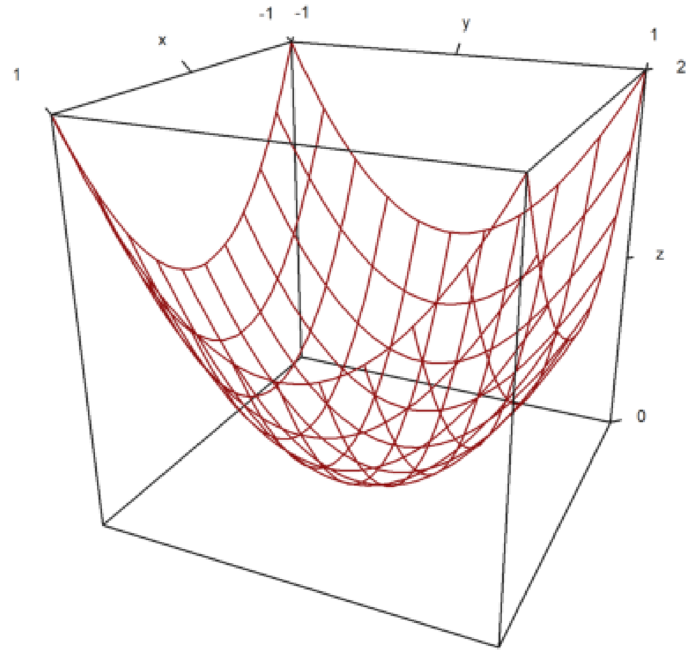
Warna 0 memberikan efek pelangi khusus.

```
>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):
```



10) Menggambar Diagram Batang Tiga Dimensi

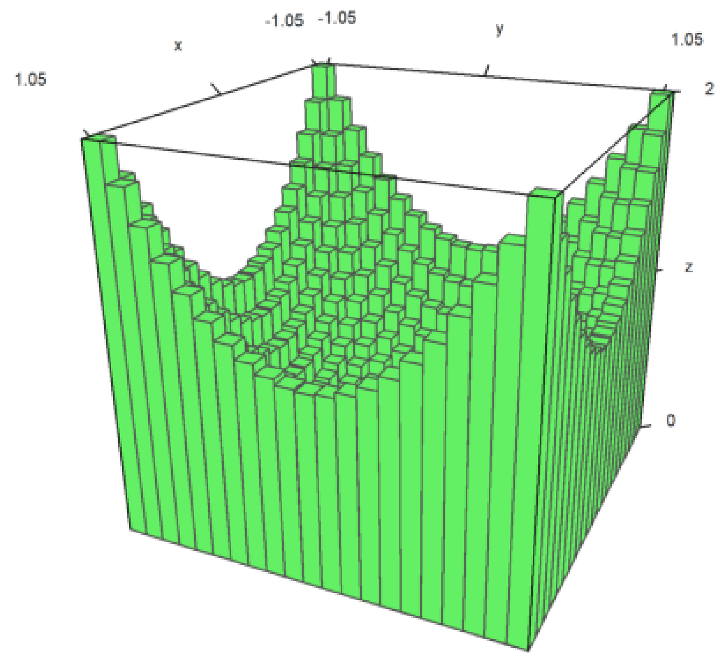
Bar plots/plot batang juga dimungkinkan. Untuk itu, kita harus menyediakannya

- x: vektor baris dengan n+1 elemen
- y: vektor kolom dengan n+1 elemen
- z: matriks nilai nxn.

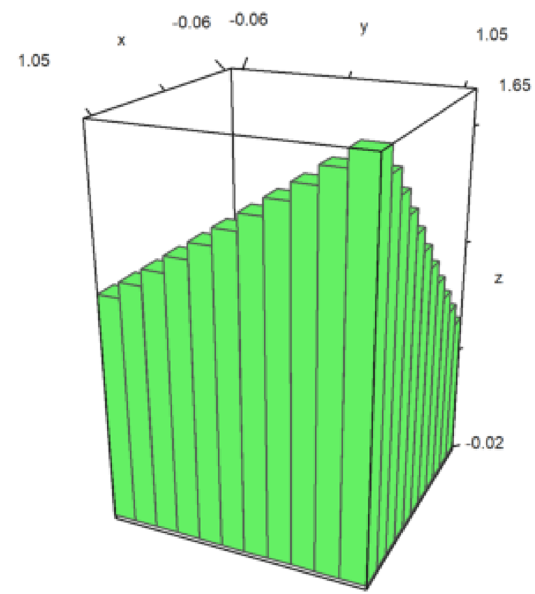
z bisa lebih besar, tetapi hanya nilai nxn yang akan digunakan.

Dalam contoh ini, pertama-tama kita menghitung nilainya. Kemudian kita sesuaikan x dan y, sehingga vektor-vektornya berpusat pada nilai yang digunakan.

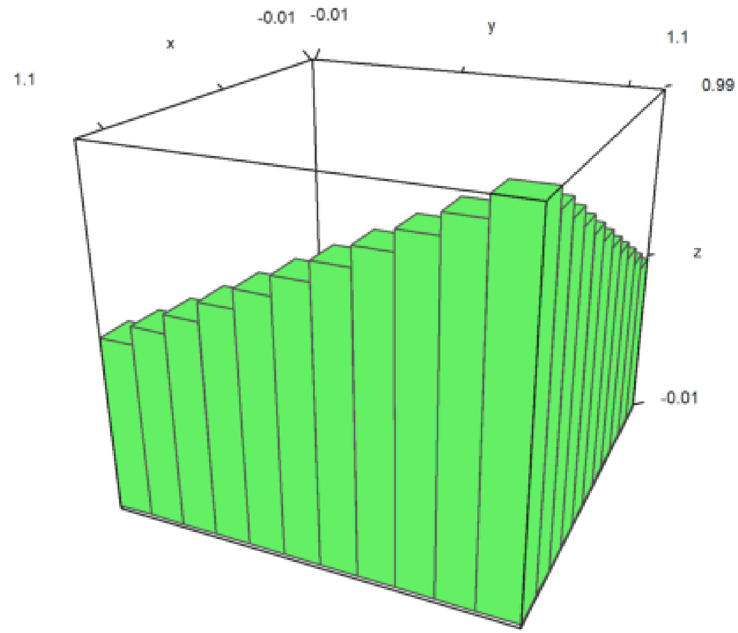
```
>x=-1:0.1:1; y=x'; z=x^2+y^2; ...
>xa=(x|1.1)-0.05; ya=(y_1.1)-0.05; ...
>plot3d(xa,ya,z,bar=true):
```




```
>x=-0.01:0.1:1; y=x'; z=x+2/3*y; ...
>xa=(x|1.1)-0.05; ya=(y_1.1)-0.05; ...
>plot3d(xa,ya,z,bar=true):
```

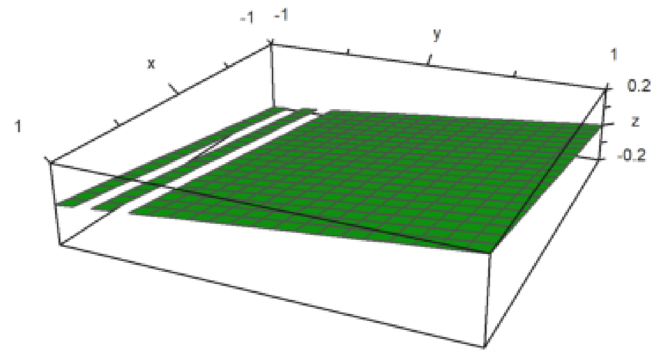



```
>x=-0.01:0.1:1; y=x'; z=1/2*x+1/2*y; ...
>xa=(x|1.1); ya=(y_1.1); ...
>plot3d(xa,ya,z,bar=true):
```

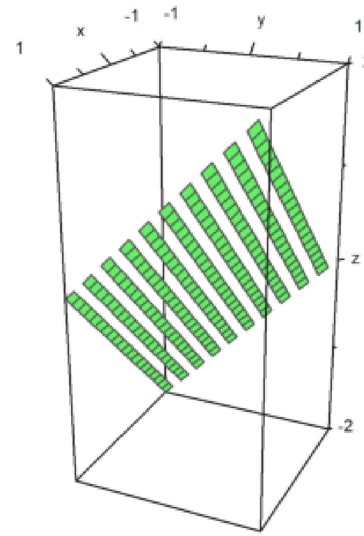


Dimungkinkan untuk membagi plot suatu permukaan menjadi dua bagian atau lebih.

```
>x=-1:0.1:1; y=x'; z=1/10*x+1/10*y; d=zeros(size(x)); ...
>plot3d(x,y,z,disconnect=2:2:5):
```

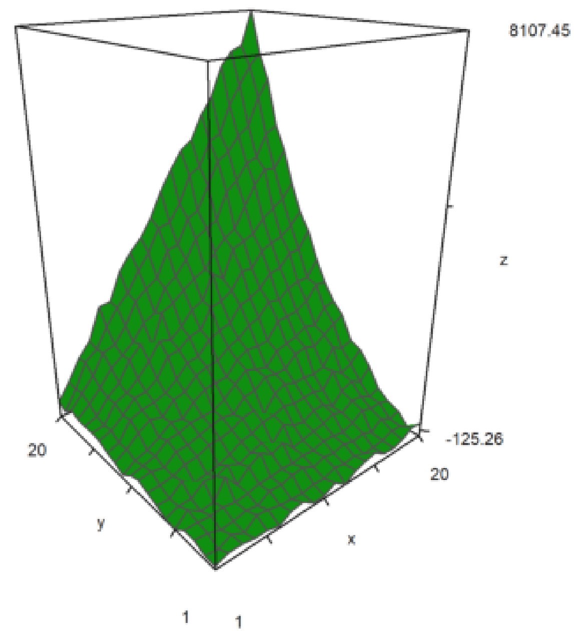



```
>x=-1:0.1:1; y=x'; z=x+y; d=zeros(size(x)); ...
>plot3d(x,y,z,disconnect=2:2:20):
```

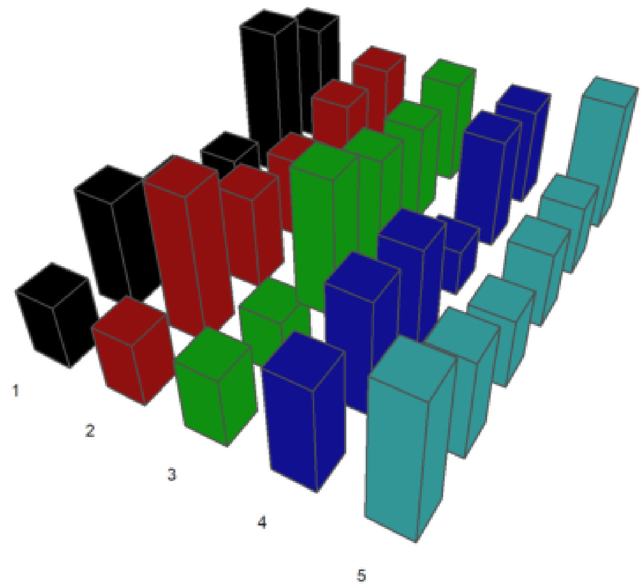


Jika memuat atau menghasilkan matriks data M dari file dan perlu memplotnya dalam 3D, Anda dapat menskalakan matriks ke [-1,1] dengan skala(M), atau menskalakan matriks dengan >zscale. Hal ini dapat dikombinasikan dengan faktor penskalaan individual yang diterapkan sebagai tambahan.

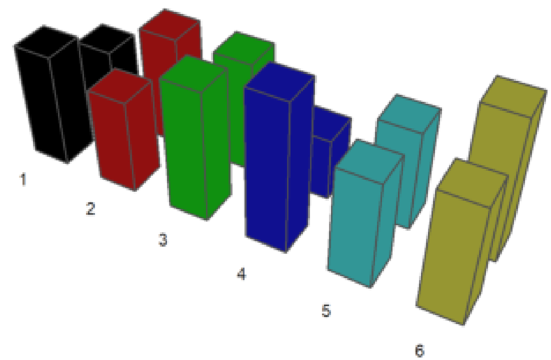
```
>i=1:20; j=i'; ...
>plot3d(i*j^2+100*normal(20,20),>zscale,scale=[1,1,1.5],angle=-40°,zoom=1.8):
```



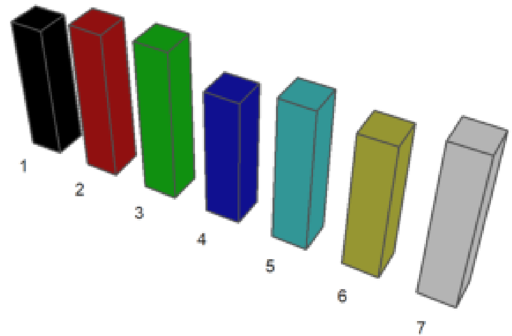

```
>Z=intrandom(5,100,6); v=zeros(5,6); ...
>loop 1 to 5; v[#]=getmultiplicities(1:6,Z[#]); end; ...
>columnsplot3d(v',scols=1:5,ccols=[1:5]):
```




```
>Z=intrandom(6,100,6); v=zeros(6,2); ...
>loop 1 to 6; v[#]=getmultiplicities(1:2,Z[#]); end; ...
>columnsplot3d(v',scols=1:6,ccols=[1:6]):
```

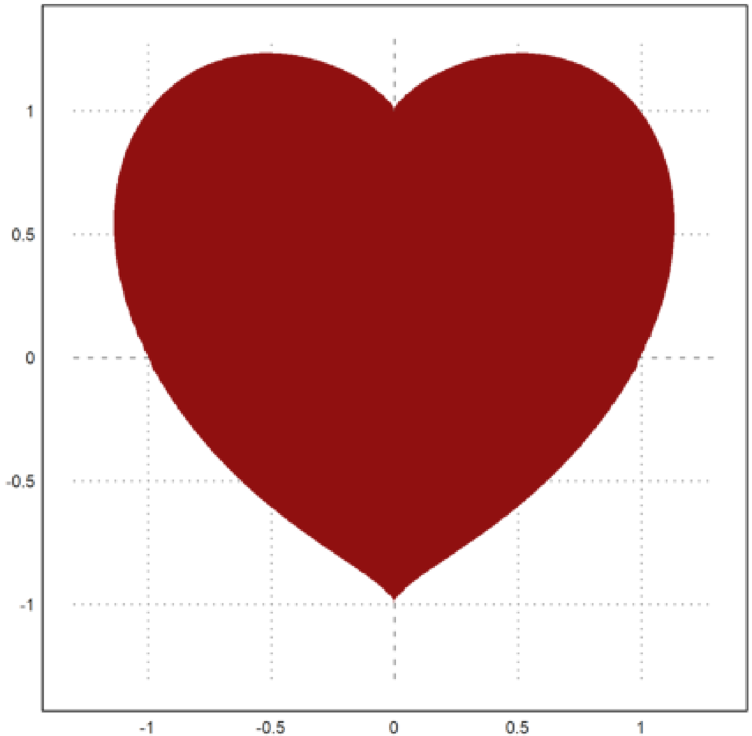



```
>Z=intrandom(7,1000,6); v=zeros(7,1); ...
>loop 1 to 7; v[#]=getmultiplicities(1:1,Z[#]); end; ...
>columnsplot3d(v',scols=1:7,ccols=[1:7]):
```



11) Menggambar 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 jantung di sekitar sumbu y. Berikut adalah ungkapan, yang mendefinisikan hati:

$$f(x,y)=(x^2+y^2-1)^3-x^2.y^3.$$

Selanjutnya kita atur

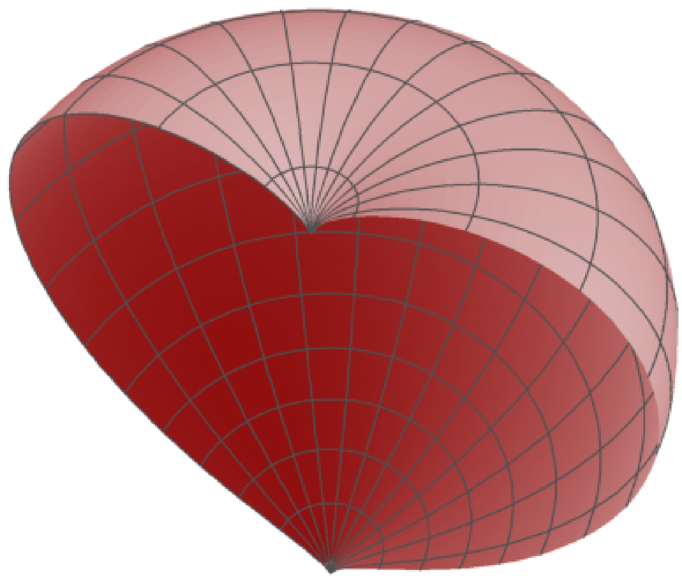
$$x=r.cos(a),\quad y=r.sin(a).$$

```
>function fr(r,a) &= ekspresi with [x=r*cos(a),y=r*sin(a)] | trigreduce; $fr(r,a)
```

$$\left(r^2-1\right)^3+\frac{\left(\sin\left(5\,a\right)-\sin\left(3\,a\right)-2\,\sin a\right)\,r^5}{16}$$

Hal ini memungkinkan untuk mendefinisikan fungsi numerik, yang memecahkan r, jika a diberikan. Dengan fungsi itu 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°):
```

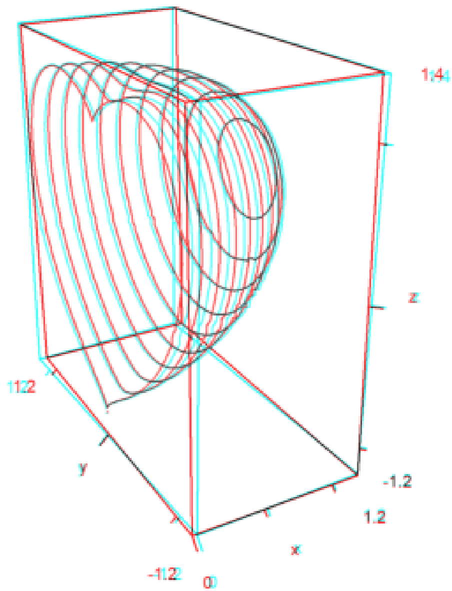


Berikut ini adalah plot 3D dari gambar di atas yang diputar di sekitar sumbu z. Kami mendefinisikan fungsi, yang menggambarkan objek.

```
>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=[10,60,60],>anaglyph):
```



12) Menggambar Grafik 3D dengan Povray di EMT

Menggambar Povray Dengan bantuan file Euler povray.e, Euler dapat menghasilkan file Povray. Hasilnya sangat bagus untuk dilihat.

Untuk dapat menjalankan sintaks dalam povray perlu menginstal Povray (32bit atau 64bit) dari http://www.povray.org/, dan meletakkan sub-direktori "bin" dari Povray ke pathway, atau mengatur variabel "defaultpovray" dengan path lengkap yang menunjuk ke "pvengine.exe".

Interface Povray dari Euler menghasilkan file Povray di direktori home pengguna, dan memanggil Povray untuk mengurai 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 buku catatan. Untuk membersihkan file-file ini, gunakan povclear().

Sintaks yang digunakan untuk menjalankan povray adalah pov3d. Fungsi pov3d memiliki komponen 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 gambar 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 gambar. Terakhir, akhiri file dengan povend(). Secara default, raytracer akan dimulai, dan PNG akan dimasukkan ke dalam notebook Euler.

Fungsi objek memiliki parameter yang disebut "look", yang membutuhkan string dengan kode Povray untuk tekstur dan hasil akhir objek. Fungsi povlook() dapat digunakan untuk menghasilkan string ini. Ini memiliki parameter untuk warna, transparansi, Phong Shading dll.

Lingkup Povray memiliki sistem koordinat lain. Interface ini menerjemahkan semua koordinat ke sistem Povray. Jadi Anda dapat terus berpikir dalam sistem koordinat Euler dengan z menunjuk vertikal ke atas, dan x,y,z sumbu dalam arti tangan kanan.
Anda perlu memuat file povray.

```
>load povray;
```

Pastikan, direktori bin Povray ada di path. Jika tidak, edit variabel berikut sehingga berisi path ke povray yang dapat dieksekusi.

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

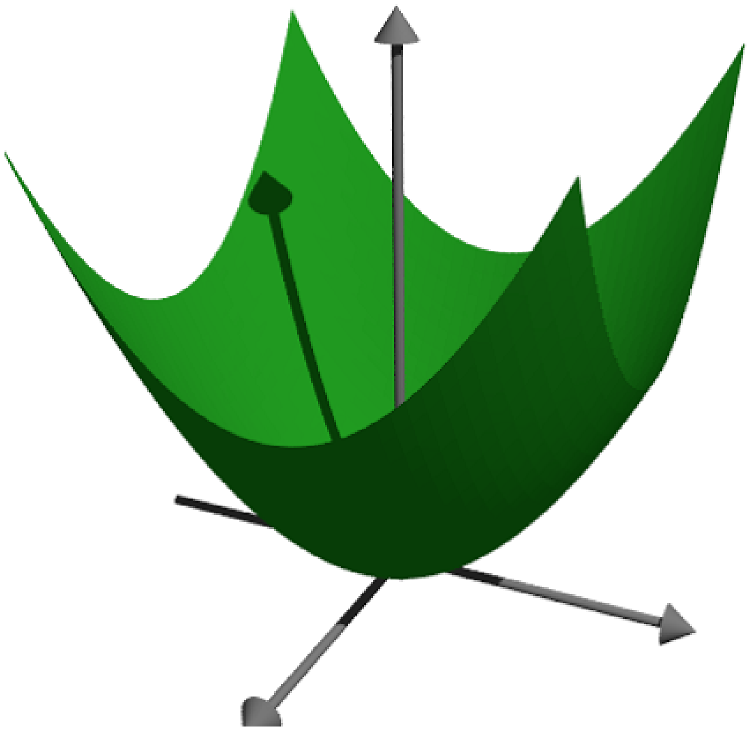
```
C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe
```

Contoh Penggunaan

Akan diberikan contoh sederhana penggunaan povray pada EMT
Perintah berikut menghasilkan file povray di direktori pengguna dan menjalankan Povray untuk ray tracing file ini.

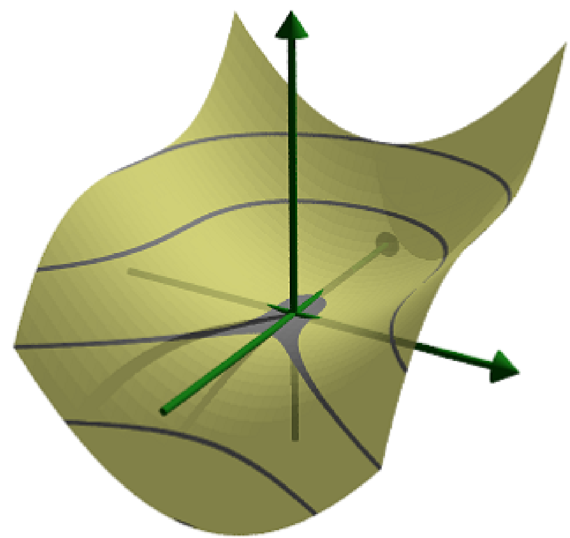
Jika memulai perintah berikut, GUI Povray akan terbuka, menjalankan file, dan menutup secara otomatis. Karena alasan keamanan, akan ditanya, apakah ingin mengizinkan file exe untuk dijalankan. Agar pertanyaan tersebut tidak muncul lagi bisa dipilih batal.

```
>pov3d("x^2+y^2",zoom=4);
```

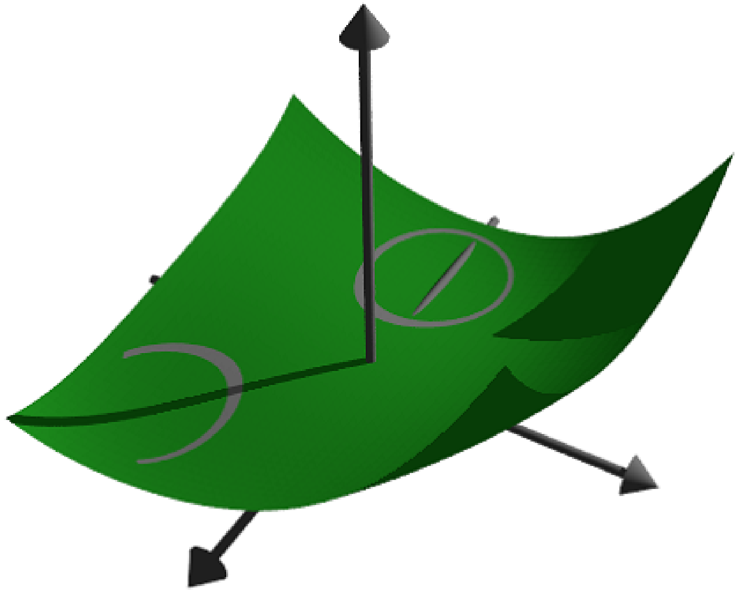


hasil visualisasi fungsi dapat dibuat menjadi transparan dan menambahkan hasil akhir lainnya.

```
> pov3d("(x^2+y^3)",axiscolor=green,angle=30°, ...
>  look=povlook(yellow,0.2),level=-1:0.5:1,zoom=3);
```



```
>pov3d("((x-1)^2+(y+1)^2)*((x+1)^2+y^2)/40",r=1.5, ...
>  angle=120°,level=1/40,dlevel=0.005,light=[-1,1,1],height=45°,n=50, ...
>  <fscale,zoom=3.8);
```



Contoh-contoh di atas tadi merupakan visualisasi permukaan fungsi dengan menggunakan sintaks pov3d. Untuk menghasilkan objek dalam povray perlu ditulis menjadi file povray.

Untuk menghasilkan output dimulai dengan povstart()

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

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

```
>povstart(zoom=3.5)
>c1=povcylinder(-povx,povx,1,povlook(orange)); ...
>c2=povcylinder(-povy,povy,1,povlook(yellow)); ...
>c3=povcylinder(-povz,povz,1,povlook(lightblue));
```

Di atas telah didefinisikan tiga silinder yang disimpan dalam string di Euler. Fungsi povx(), povy(), dll. hanya mengembalikan vektor
,0
yang dapat digunakan sebagai gantinya.

```
>c1
```

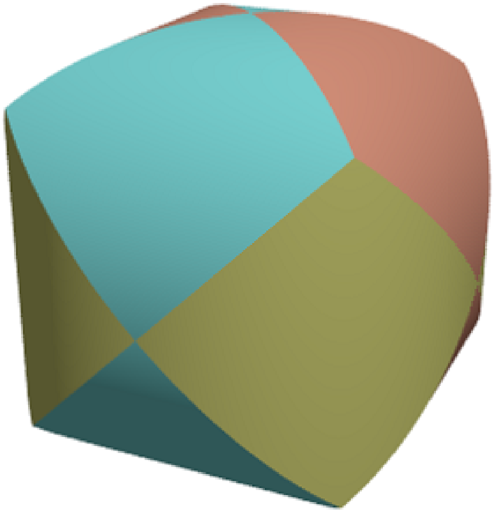
```
cylinder { <-1,0,0>, <1,0,0>, 1
 texture { pigment { color rgb <0.941176,0.509804,0.392157> }  }
 finish { ambient 0.2 }
 }
```

Akan ditambahkan tekstur ke objek dengan tiga warna berbeda yaitu orange, yellow, dan lightblue. Untuk menamahkan tekstur ini dapat digunakan sintaks povlook(), yang mengembalikan string dengan kode Povray yang relevan. Selain menambahkan warna, ditambahkan juga transparansi dan cahaya.

```
>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 }
```

```
>writeln(povintersection([c1,c2,c3]));
>povend;
```



Contoh Lain

Akan ditampilkan fungsi untuk membuat sebuah donat

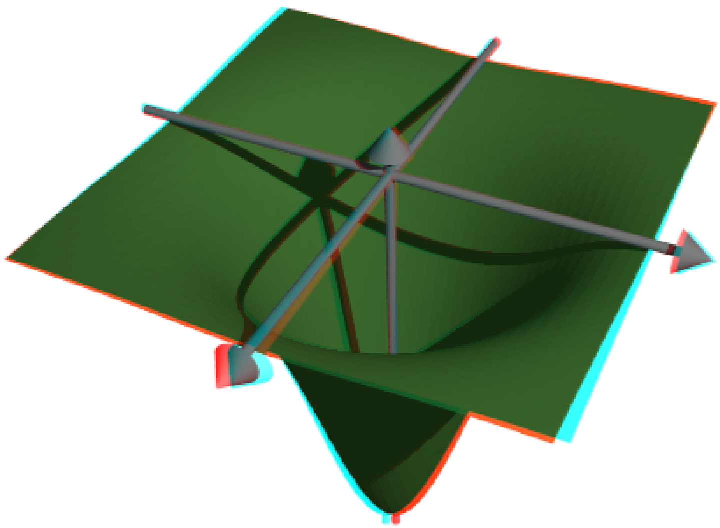
```
>povstart(angle=0,height=45°); //height untuk menampilkan fungsi dengan suatiu derajat tertentu
>function povdonat (r1,r2,look="") := "torus {"+r1+","+r2+look+"}"; //fungsi untuk menampilkan sebua
>writeln(povobject(povdonat(1,0.5),povlook(lightblue,>phong),xrotate(90°)));
>povend();
```

Untuk menghasilkan anaglyph untuk kacamata merah/sian, Povray harus berjalan dua kali dari posisi kamera yang berbeda. Ini menghasilkan dua file Povray dan dua file PNG, yang dimuat dengan fungsi loadanaglyph().

Tentu saja, Anda memerlukan kacamata merah/sian untuk melihat contoh berikut dengan benar.

Fungsi pov3d() memiliki sakelar sederhana untuk menghasilkan anaglyphs.

```
>pov3d("-exp(-x^2-y^2)/2",r=2,height=45°,>anaglyph, ...
>  center=[0,0,0.5],zoom=3.5);
```



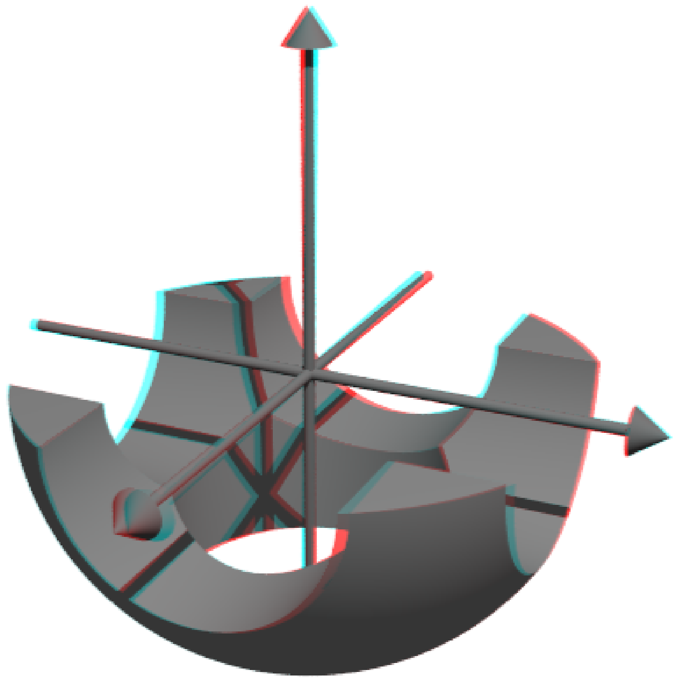
Jika Anda membuat adegan dengan objek, Anda perlu menempatkan generasi adegan ke dalam fungsi, dan menjalankannya dua kali dengan nilai yang berbeda untuk parameter anaglyph.

```
>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. Parameternya seperti di povstart() dan povend() digabungkan.

```
>povanaglyph("myscene",zoom=4.5);
```



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.

$$((x^2+y^2-c^2)^2+(z^2-1)^2)*((y^2+z^2-c^2)^2+(x^2-1)^2)*((z^2+x^2-c^2)^2+(y^2-1)^2)=d$$

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

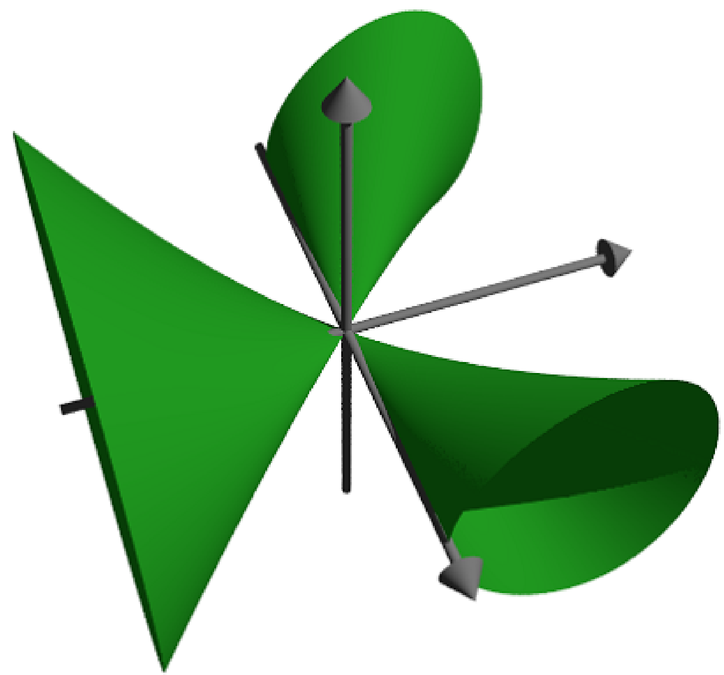
```
C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe
```

```
>povstart(angle=25°,height=10°);
>writeln(povsurface("pow(x,2)+pow(y,2)*pow(z,2)-1",povlook(blue),povbox(-2,2,"")));
>povend();
```

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

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

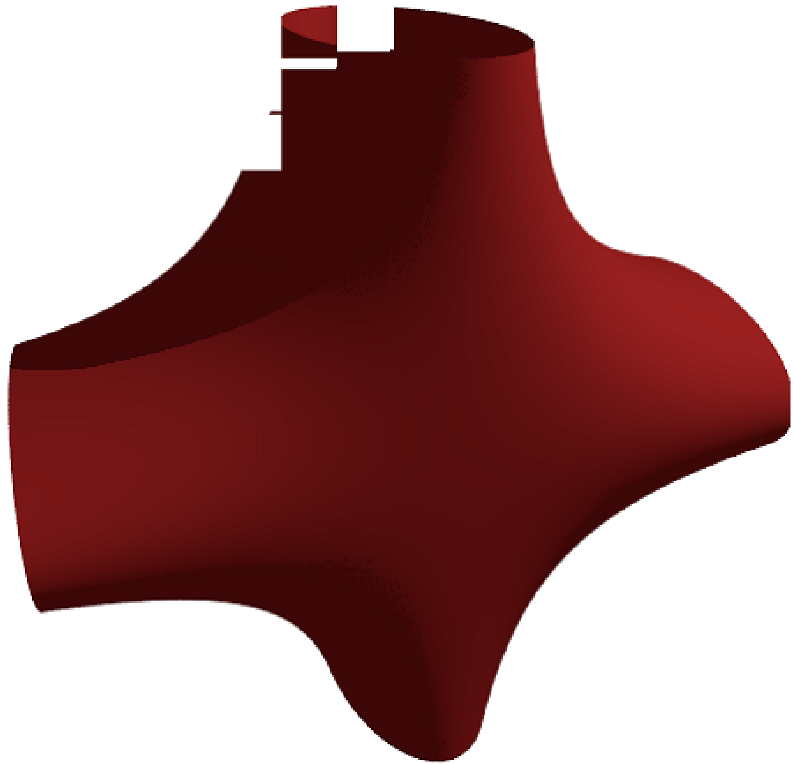
```
>povstart(angle=70°,height=50°,zoom=4);
>writeln(povsurface("pow(x,2)*y-pow(y,3)-pow(z,2)",povlook(green))); ...
>writeAxes(); ...
>povend();
```



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

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

```
>povstart(angle=70°,height=30°);
>writeln(povsurface("pow(x,2)+pow(y,2)*pow(z,2)-1",povlook(red),povbox(-2,2,"")));
>povend();
```



Contoh lain

```
>povstart(angle=45, height=100);
>defaultpovray="C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe"
```

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

```
>writeln(povsurface("pow(x,2)+pow(y,3)*pow(z,2)-1", povlook(blue),povbox(-25,4,"")));
>povend();
```

Alih-alih fungsi, kita dapat memplot dengan koordinat. Seperti pada plot3d, kita membutuhkan tiga matriks untuk mendefinisikan objek.

Dalam contoh kita memutar fungsi di sekitar sumbu z.

```
>function f(x) := x^3-x+1; ...
>x=-1:0.01:1; t=linspace(0,2pi,8)'; ...
>Z=x; X=cos(t)*f(x); Y=sin(t)*f(x); ...
>pov3d(X,Y,Z,angle=40°,height=20°,axis=0,zoom=4,light=[10,-5,5]);
```

Dalam contoh berikut, kami 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=5,axis=0,add=povsphere([0,0,0.5],0.1,povlook(green)), ...
>  w=500,h=300);
```

Dengan metode bayangan canggih dari Povray, sangat sedikit titik yang dapat menghasilkan permukaan yang sangat halus. Hanya di perbatasan dan dalam bayang-bayang triknya mungkin menjadi jelas.

Untuk ini, kita perlu menambahkan vektor normal di setiap titik matriks.

```
>Z &= x^2*y^3
```

$$x^{2} y^{3}$$

Persamaan permukaannya adalah [x,y,Z]. Kami menghitung dua turunan ke x dan y ini dan mengambil produk silang sebagai normal.

```
>dx &= diff([x,y,Z],x); dy &= diff([x,y,Z],y);
```

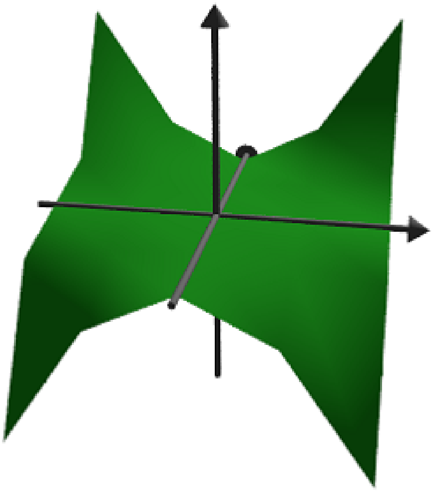
Kami mendefinisikan normal sebagai produk silang dari turunan ini, dan mendefinisikan fungsi koordinat.

```
>N &= crossproduct(dx,dy); NX &= N[1]; NY &= N[2]; NZ &= N[3]; N,
```

$$[- 2 x y^{3}, - 3 x^{2} y^{2}, 1]$$

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 ini adalah simpul Trefoil yang dilakukan oleh A. Busser di Povray. Ada versi yang ditingkatkan dari ini dalam contoh.

See: Contoh\Trefoil Simpul | Simpul trefoil

Untuk tampilan yang bagus dengan tidak terlalu banyak titik, kami menambahkan vektor normal di sini. Kami menggunakan Maxima untuk menghitung normal bagi kami. Pertama, ketiga fungsi koordinat sebagai ekspresi simbolik.

```
>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 kedua vektor turunan ke x dan y.

```
>dx &= diff([X,Y,Z],x); dy &= diff([X,Y,Z],y);
```

Sekarang normal, yang merupakan produk silang dari dua turunan.

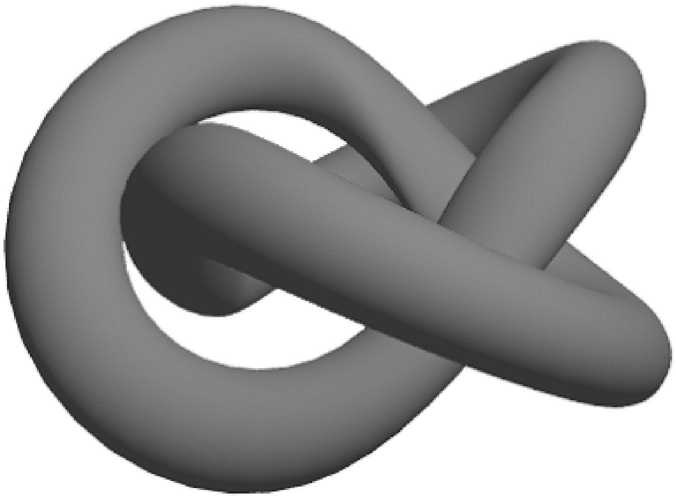
```
>dn &= crossproduct(dx,dy);
```

Kami sekarang mengevaluasi semua ini secara numerik.

```
>x:=linspace(-%pi,%pi,40); y:=linspace(-%pi,%pi,100)';
```

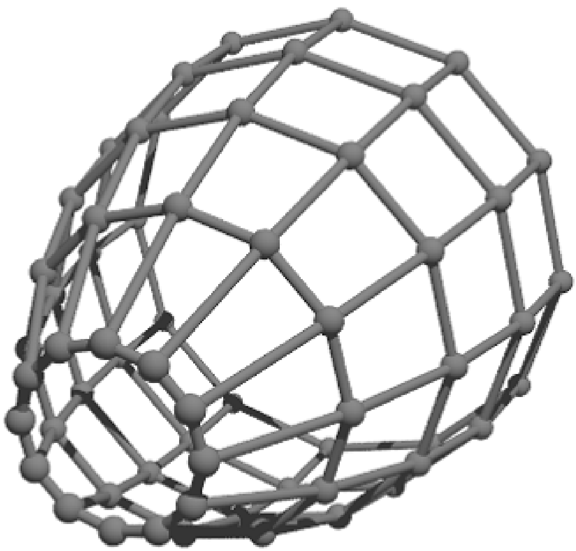
Vektor normal adalah evaluasi dari ekspresi simbolik dn[i] untuk i=1,2,3. Sintaks untuk ini adalah &"expression"(parameters). Ini adalah alternatif dari metode pada contoh sebelumnya, di mana kita mendefinisikan ekspresi simbolik NX, NY, NZ terlebih dahulu.

```
>pov3d(X(x,y),Y(x,y),Z(x,y),axis=0,zoom=5,w=450,h=350, ...
>  <shadow,look=povlook(gray), ...
>  xv=&"dn[1]"(x,y), yv=&"dn[2]"(x,y), zv=&"dn[3]"(x,y));
```



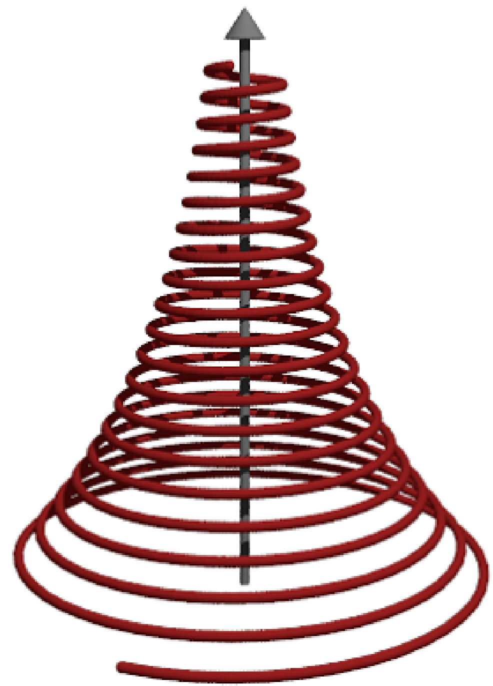
Kami juga dapat menghasilkan grid dalam 3D.

```
>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();
```



With povgrid(), curves are possible.

```
>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();
```

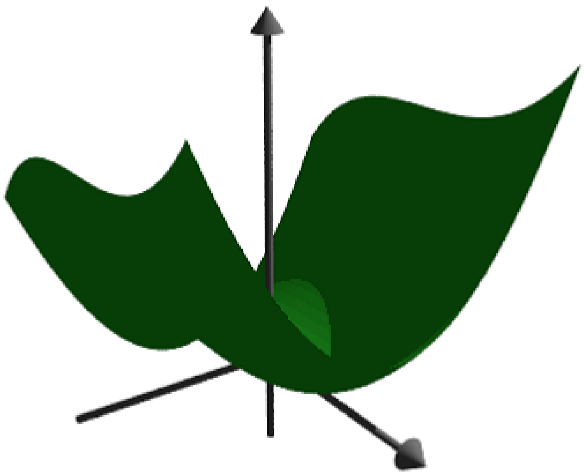


1. Buatlah plot 3D dari fungsi

$$f(x, y) = x^3 + 3y^2$$

dengan zoom 3 dan angle 55 derajat menggunakan povray

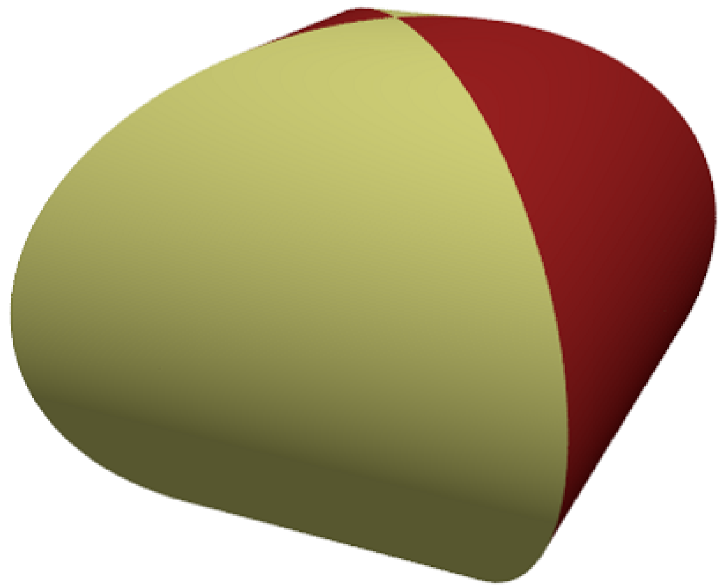
```
>pov3d("x^3+3*y^2",zoom=3,angle=55°);
```



2. Buatlah gabungan 2 silinder dengan fungsi povx() berwarna merah dan povz() berwarna kuning dan zoom 4

```
>povstart(zoom=4)
>c1 = povcylinder (-povx,povx,1,povlook(red));
>c2 = povcylinder (-povz,povz,1,povlook(yellow));
```

```
>writeln(povintersection([c1,c2]));
>povend();
```



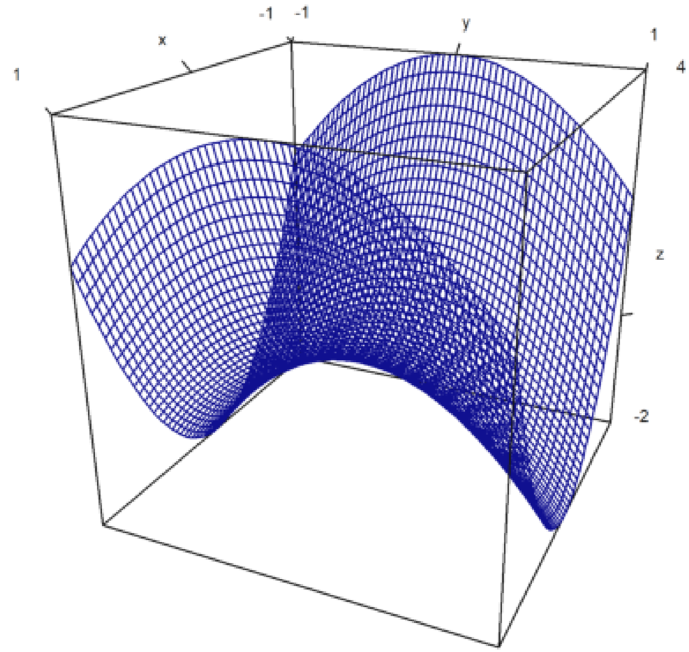
```
>
```

3. Buatlah grafik 3D dari fungsi kuadrat berikut ini dengan parameter tambahan:

$$z = 4x^2 - 2y^2$$

Tampilkan grafik tersebut dengan transparent, dan menggunakan grid dengan resolusi 50, dengan warna biru pada garis di plot tersebut

```
>plot3d("4*x^2-2*y^2",>transparent,grid=50,wirecolor=blue):
```


x
-1
-1
y
1
1
4
z
-2

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).

Fungsi

Fungsi adalah suatu relasi yang menghubungkan setiap anggota dari suatu himpunan (domain) dengan tepat satu anggota himpunan lain (kodomain). Secara sederhana, fungsi dapat dipandang sebagai mesin yang menerima input (nilai x) dan menghasilkan output (nilai y).

$$f : A \to B$$

Sifat-sifat Fungsi

1. Injektif (satu-satu)
Fungsi injektif adalah fungsi degan tiap elemen kondomain tidak memiliki lebih dari satu elemen domain. Fungsi injektif disebut juga dengan "fungsi satu-satu" karena tiap elemen kodomain hanya boleh berelasi satu kali.

2. Surjektif
Fungsi surjektif adalah fungsi dengan semua elemen kodomain berelasi dengan elemen domain. Fungsi surjektif juga disebut fungsi "on-to".

3. Bijektif
Fungsi bijektif adalah fungsi yang memenuhi sifat injektif dan surjektif. Fungsi bijektif juga disebut fungsi korespondensi satu-satu, karena elemen domain dan kodomain semuanya berelasi satu-satu.

1. Fungsi Aljabar
Fungsi aljabar adalah jenis fungsi matematika yang dibentuk dengan menggunakan operasi-operasi aljabar seperti penjumlahan, pengurangan, perkalian, pembagian, dan pangkat. Secara umum, fungsi aljabar merupakan fungsi yang dapat dinyatakan dengan ekspresi aljabar yang melibatkan variabel dan konstanta. Fungsi ini termasuk di dalam kategori fungsi elementer karena bisa dinyatakan dalam bentuk dasar.

2. Fungsi Trigonometri
Fungsi trigonometri adalah fungsi yang menghubungkan sudut suatu segitiga siku-siku dengan rasio panjang sisi-sisinya. Fungsi ini sangat penting dalam matematika, khususnya dalam geometri, analisis, dan aplikasi di bidang fisika, teknik, dan astronomi. Fungsi-fungsi trigonometri yang paling umum adalah sinus, kosinus, tangen, serta turunan dan kebalikannya seperti secan, kosekan, dan kotangen.

3. Fungsi Eksponensial
Fungsi eksponensial adalah fungsi matematika di mana variabel independen (biasanya dilambangkan dengan x) berada di eksponen. Bentuk umum fungsi eksponensial adalah:

$$f(x) = a^x$$

- a adalah bilangan positif (basis) dan tidak sama dengan 1
- x adalah variabel eksponen

Fungsi eksponensial yang paling umum adalah fungsi eksponensial alami dengan basis bilangan euler (e = 2.718). Fungsi ini sering digunakan dalam berbagai aplikasi ilmiah yang dinyatakan sebagai:

$$f(x) = e^x$$

4. Fungsi Logaritma
Fungsi logaritma adalah kebalikan atau invers dari fungsi eksponensial. Fungsi logaritma dinyatakan sebagai:

$$f(x) = \log a(x)$$

5. Komposisi Fungsi
Komposisi fungsi adalah operasi matematika di mana dua fungsi digabungkan sehingga output dari satu fungsi menjadi input dari fungsi yang lain. Diketahui f gan g dua fungsi sebarang maka fungsi komposisi f dan g ditulis

$$gof$$

didefinisikan sebagai:

$$(gof)(x) = g(f(x))$$

dengan f dikerjakan terlebih dahulu daripada g.

Mendefinisikan Fungsi

Terdapat beberapa cara mendefinisikan fungsi pada EMT, yakni:

- Menggunakan format nama_fungsi := rumus fungsi (fungsi numerik),
- Menggunakan format nama_fungsi &= rumus fungsi (fungsi simbolik, namun dapat dihitung secara numerik),
- Menggunakan format nama_fungsi &&= rumus fungsi(fungsi simbolik murni, tidak dapat dihitung langsung),
- Fungsi sebagai program EMT (Mendefinisikan fungsi yang lebih kompleks dengan beberapa langkah atau percabangan)

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 karena bukan fungsi numerik
```

```
50.3833049952
```

$$f(x) = 3x^3 + e^{\cos(x)}.$$

```
>function f(x) := 3*x^3+exp(cos(x))
>f(1)
```

```
4.71652569955
```

```
>f(20)
```

```
24001.5039306
```

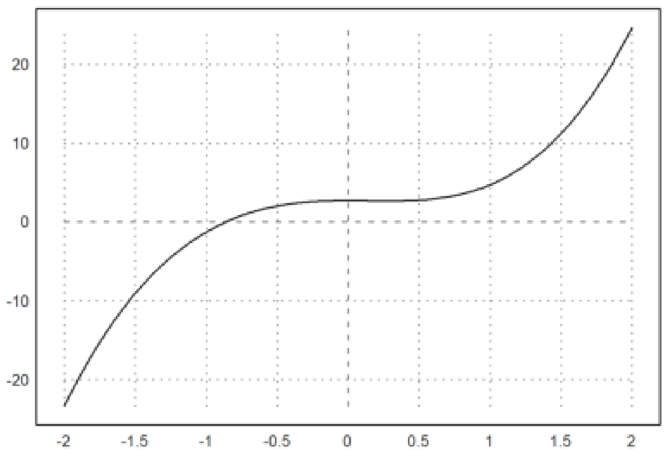
```
>f(-2:2)
```

```
[-23.3404,  -1.28347,  2.71828,  4.71653,  24.6596]
```

```
>f(-2)+f(2)
```

```
1.31916682485
```

```
>aspect(1.5); plot2d("f(x)",-2,2):
```



Berikutnya kita definisikan fungsi:

$$g(x) = \frac{\sqrt{x^2 - 3x}}{x + 1}$$

```
>function g(x) := sqrt(x^2-3*x)/(x+1)
>g(1)
```

```
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) ...
    ^
```

Floating point error karena untuk x=1, g(x) akan bernilai imajiner yaitu:

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

```
>g(5)
```

```
0.527046276695
```

```
>g(20)
```

```
0.878051853076
```

```
>g(21)
```

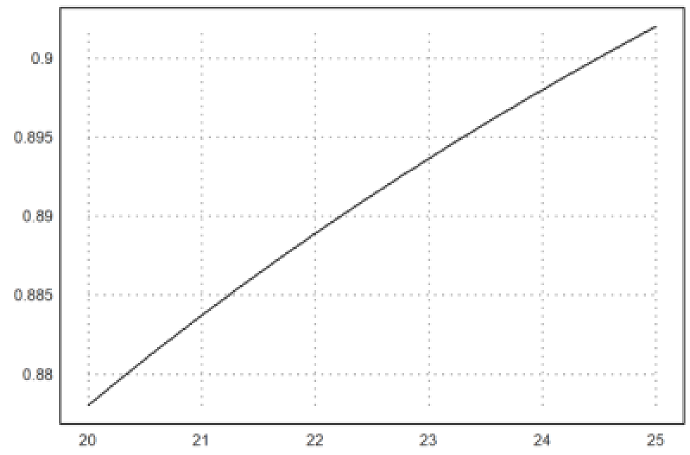
```
0.883737367965
```

```
>g(20:25)
```

```
[0.878052,  0.883737,  0.888915,  0.89365,  0.897998,  0.902003]
```

Silakan Anda plot grafik fungsi di atas!

```
>aspect(1.5); plot2d("g(x)",20,25):
```




```
>f(g(5)) // komposisi fungsi
```

```
2.81254111152
```

```
>g(f(5))
```

```
0.993366510057
```

```
>function h(x) := f(g(x)) // definisi komposisi fungsi
>h(5) // sama dengan f(g(5))
```

```
2.81254111152
```

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.

```
>f(0:10) // nilai-nilai f(0), f(1), f(2), ..., f(10)
```

```
[2.71828,  4.71653,  24.6596,  81.3716,  192.52,  376.328,  650.612,
1031.13,  1536.86,  2187.4,  3000.43]
```

```
>fmap(0:10) // sama dengan f(0:10), berlaku untuk semua fungsi
```

```
[2.71828,  4.71653,  24.6596,  81.3716,  192.52,  376.328,  650.612,
1031.13,  1536.86,  2187.4,  3000.43]
```

```
>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

$$f(x) = \begin{cases} x^3 & x > 0 \\ x^2 & x \leq 0. \end{cases}$$

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
```

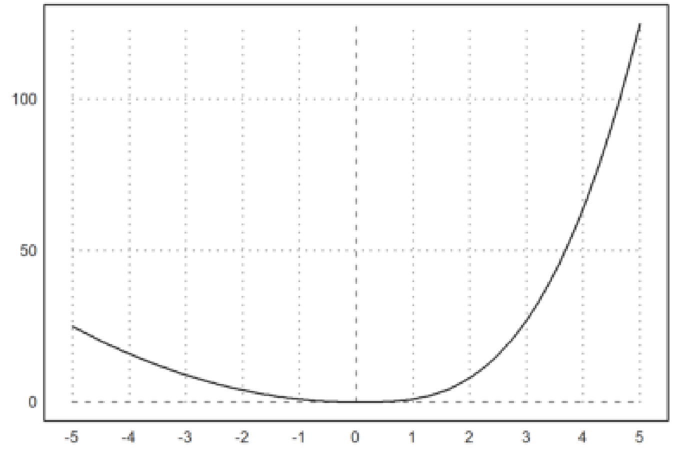
```
>f(-2)
```

```
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
```

$$2\,e^{a}$$

```
>function a:=12
>f(a)
```

```
296.826318205
```

```
>f(E) // nilai fungsi berupa bilangan desimal
```

```
30.308524483
```

```
>$f(E), $float(%)
```

$$30.30852448295852$$

```
>function g(x) &= 3*x+1
```

```
                                   3 x + 1
```

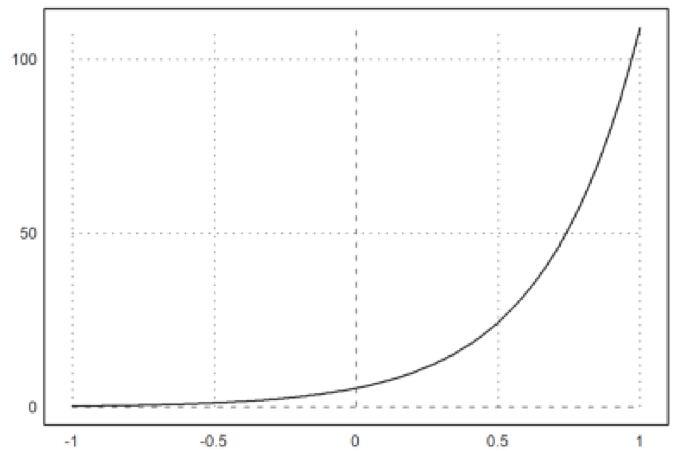
```
>g(50)
```

```
  151
```

```
>function h(x) &= f(g(x)) // komposisi fungsi
```

```
                                     3 x + 1
                                  2 E
```

```
>plot2d("h(x)",-1,1):
```



Contoh Fungsi Aljabar

1.Diberikan fungsi:

$$f(x) = 6x^3 + 12\sqrt{x}$$

Tentukan nilai f(1) dan f(4) kemudian cari nilai f(4)-2f(1)
Gambarkan juga grafiknya di f(1:4)!

```
>function f(x) := 6*x^3+12*sqrt(x)
>f(1)
```

```
18
```

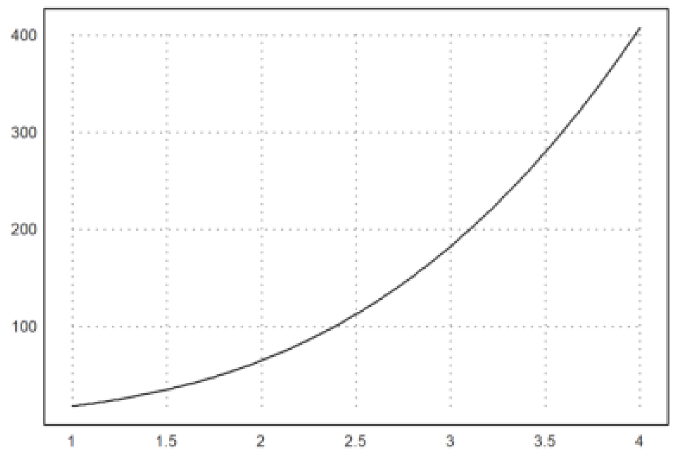
```
>f(4)
```

```
408
```

```
>f(4)-2*f(1)
```

```
372
```

```
>aspect(1.5); plot2d("f",1,4):
```



2. Diberikan fungsi:

$$k(x,y) = 12x^3 + 4y^2 - 3x + 2$$

Tentukan nilai k(1,1)+k(2,3)!

```
>function k(x,y):= 12*x^3+4*y^2-3*x+2
>k(1,1)
```

15

```
>k(2,3)
```

```
128
```

```
>k(1,1)+k(2,3)
```

```
143
```

3. Diketahui suatu fungsi f(x,y,z) diddefinisikan sebagai betrikut

$$f(x, y, z) = (3x - 1) + \sqrt{x^3 - y^2 - z + 6}$$

Tentukan nilai f(3,4,5)!

```
>function f(x,y,z):= (3*x-1)+ sqrt(x^3-y^2-z+6)
>f(3,4,5)
```

```
11.4641016151
```

Contoh Fungsi Trigonometri

1. Suatu fungsi trigonometri didefinisikan sebagai:

$$f(x) = \frac{tan(x)}{2x + 3x^2}$$

Tentukan nilai dari f(6)-f(7)!

```
>function f(x):= tan(x)/2*x+3*x^2
>f(6)-f(7)
```

```
-42.9230865137
```

2. Diberikan fungsi:

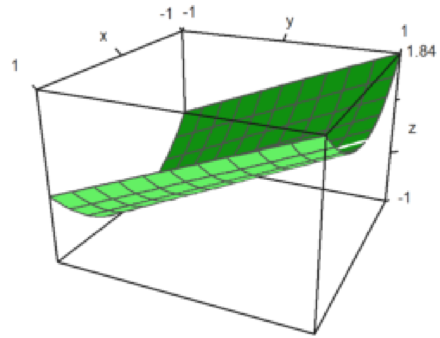
$$f(x, y) = \frac{x^2 + sin(y)}{xy}$$

Tentukan nilai f(4)!

```
>function f(x,y) := x^2+sin(x)/x*y
>f(4,2)
```

```
15.6215987523
```

```
>plot3d("f",-4,4):
```



3. Diberikan suatu fungsi

$$f(x, y, z) = \frac{sin(x)}{cos(z) + sin(x)} + tan(y)$$

tentukan nilai f(x,y,z) bila x=1, y=2, dan z=-3!

```
>function f(x,y,z)&= sin(x)/(sin(x)+cos(z))+tan(y)
```

```
    sin(x)
--------------- + tan(y)
cos(z) + sin(x)
```

```
>f(1,2,-3)
```

```
-7.85069041214
```

Contoh Fungsi Eksponensial

1. Diberikan sebuah fungsi f(x) sebagai berikut:

$$f(x) = 3^x$$

Tentukan nilai f(x+1)-f(x)!

```
>function f(x):= 3^x
>$f(x+1)
```

```
Maxima said:
Too few arguments supplied to f(x,y,z); found: [x+1]
 -- an error. To debug this try: debugmode(true);

Error in:
 $f(x+1) ...
       ^
```

```
>$f(x)
```

```
Maxima said:
Too few arguments supplied to f(x,y,z); found: [x]
 -- an error. To debug this try: debugmode(true);

Error in:
 $f(x) ...
     ^
```

```
>$f(x+1)-f(x)
```

```
Maxima said:
```

```
Too few arguments supplied to f(x,y,z); found: [x+1]
 -- an error. To debug this try: debugmode(true);

Error in:
 $f(x+1)-f(x) ...
            ^
```

2. Fungsi f(m) didefinisikan sebagai:

$$f(m) = 27 + 8^m$$

Tentukan nilai dari f(3), 2f(5), dan f(3)/2f(5)

```
>function f(m)&=27+8^m;
>$f(3)
```

$$539$$

```
>$2*f(5)
```

$$65590$$

```
>$f(3)/(2*f(5))
```

$$\frac{77}{9370}$$

3. Diketahui nilai y dari fungsi di bawah ini adalah 1

$$f(x) = 2^{4x-12y}$$

Tentukan nilai f(3), f(8), dan buat grafiknya di f(3:5)!

```
>function f(x):=2^(4*x-12*y)
>function y:=1
>f(8)
```

```
[4.29497e+09,  1.51745e+09,  5.47254e+08,  2.04699e+08,  8.06415e+07,
3.39465e+07,  1.54748e+07,  7.73187e+06,  4.27913e+06,  2.64671e+06,
1.8426e+06,  1.45143e+06,  1.29785e+06,  1.31903e+06,  1.52251e+06,
1.99039e+06,  2.93317e+06,  4.84062e+06,  8.87172e+06,  1.78786e+07,
3.91673e+07,  9.21022e+07,  2.293e+08,  5.9565e+08,  1.59012e+09,
4.29497e+09,  1.15549e+10,  3.04854e+10,  7.76848e+10,  1.88436e+11,
4.29142e+11,  9.06006e+11,  1.75298e+12,  3.07733e+12,  4.86012e+12,
6.85888e+12,  8.60662e+12,  9.57275e+12,  9.42643e+12,  8.22395e+12,
6.37353e+12,  4.40751e+12,  2.73678e+12,  1.53803e+12,  7.89735e+11,
3.74544e+11,  1.66061e+11,  6.97353e+10,  2.81245e+10,  1.10521e+10,
4.29497e+09,  1.67538e+09,  6.6579e+08,  2.73468e+08,  1.17719e+08,
5.38075e+07,  2.64326e+07,  1.41083e+07,  8.2603e+06,  5.34814e+06,
3.85383e+06,  3.10543e+06,  2.80654e+06,  2.84796e+06,  3.24272e+06,
4.13249e+06,  5.86929e+06,  9.23519e+06,  1.59777e+07,  3.01218e+07,
6.12433e+07,  1.3276e+08,  3.03043e+08,  7.18849e+08,  1.74781e+09,
4.29497e+09,  1.05164e+10,  2.52986e+10,  5.89768e+10,  1.31488e+11,
2.76892e+11,  5.44449e+11,  9.89291e+11,  1.64613e+12,  2.48915e+12,
3.39954e+12,  4.17463e+12,  4.59648e+12,  4.53286e+12,  4.00632e+12,
```

```
3.18112e+12,  2.27844e+12,  1.4804e+12,  8.7886e+11,  4.80821e+11,
2.44814e+11,  1.17278e+11,  5.34883e+10,  2.35191e+10,  1.01014e+10,
 ... ]
```

```
>f(3)
```

```
[4096,  1447.16,  521.902,  195.216,  76.9058,  32.3739,  14.7579,
7.37368,  4.0809,  2.5241,  1.75724,  1.38419,  1.23772,  1.25793,
1.45198,  1.89818,  2.79729,  4.61637,  8.46073,  17.0504,  37.3528,
87.8355,  218.678,  568.056,  1516.46,  4096,  11019.6,  29073.2,
74086,  179706,  409262,  864034,  1.67177e+06,  2.93477e+06,
4.63497e+06,  6.54114e+06,  8.20791e+06,  9.12928e+06,  8.98974e+06,
7.84297e+06,  6.07827e+06,  4.20333e+06,  2.61e+06,  1.46678e+06,
753150,  357193,  158368,  66504.7,  26821.6,  10540.1,  4096,
1597.77,  634.947,  260.799,  112.266,  51.3148,  25.2081,  13.4547,
7.87764,  5.10038,  3.6753,  2.96157,  2.67652,  2.71603,  3.0925,
3.94105,  5.59739,  8.80736,  15.2375,  28.7264,  58.4062,  126.61,
289.004,  685.547,  1666.85,  4096,  10029.2,  24126.6,  56244.6,
125397,  264064,  519227,  943461,  1.56987e+06,  2.37383e+06,
3.24205e+06,  3.98124e+06,  4.38354e+06,  4.32287e+06,  3.82072e+06,
3.03375e+06,  2.17289e+06,  1.41181e+06,  838146,  458547,  233472,
111845,  51010.4,  22429.6,  9633.44,  4096,  1747.52,  758.199,
338.943,  158.089,  77.849,  40.9192,  23.1851,  14.2841,  9.63889,
7.16572,  5.89402,  5.37828,  5.45006,  6.12932,  7.63297,  10.485,
15.8013,  25.9482,  46.0541,  87.5224,  176.259,  371.946,  812.67,
1815.74,  4096,  9209.93,  20380.2,  43834.2,  90548.7,  177635,
 ... ]
```

```
>f(3:5)
```

```
Cannot combine a 1x3 and a 1x1001 matrix for -!
Try "trace errors" to inspect local variables after errors.
f:
    useglobal; return 2^(4*x-12*y)
Error in:
f(3:5) ...
      ^
```

```
>aspect(1.3); plot2d("f",3,5):
```

```
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());
```

Contoh Fungsi Logaritma

1. Diberikan fungsi di bawah ini:

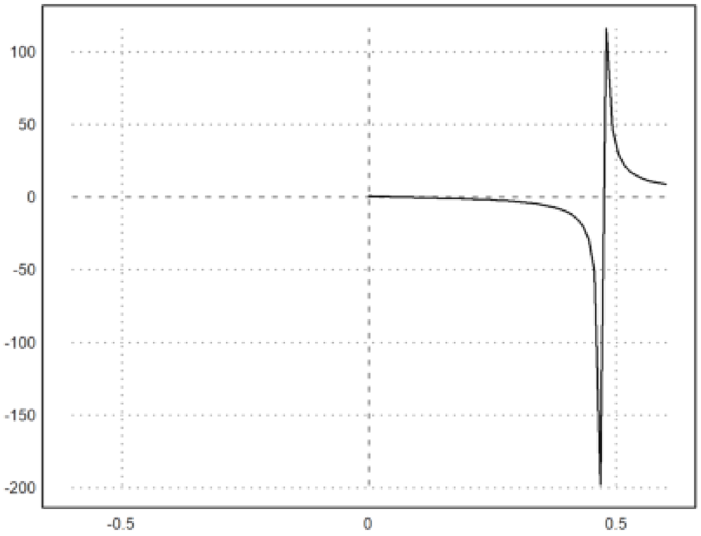
$$f(x) = e^2 \cdot \left(\frac{1}{3 + 4\log(x)} + \frac{1}{7} \right)$$

Tentukan nilai f(0.6)!

```
>function f(x):= E^2*(1/(3+4*log(x))+1/7)
>f(0.6)
```

```
8.77908249441
```

```
>plot2d("f",-0.6,0.6):
```



Contoh Komposisi Fungsi

1. Untuk fungsi:

$$f(x) = x^2 - 3x + 2$$

$$g(x) = 3x + 5$$

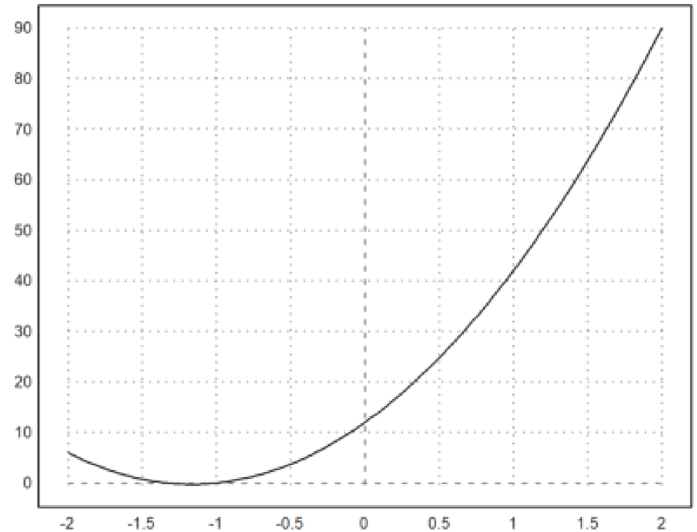
cari nilai

$$fog(-2), gof(0)$$

```
>function f(x) :=  x^2-3*x+2; $f(x)
>function g(x) := 3*x+5; $g(x)
>f(g(-2)), g(f(0))
```

```
6
11
```

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


90
80
70
60
50
40
30
20
10
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2

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.

1. Diberikan fungsi:

$$b(x) = \frac{12x^3}{7x^2 - 3x}$$

Cari nilai b(1), vektor b(1:4), dan buatkan grafiknya!

```
>
>
```

2. Diketahui suatu fungsi f(x) sebagai berikut:

$$f(x, y) = \frac{3(x-1)(y-2)}{2} + \frac{(y-2)(x-3)}{2} - 2(x-1)(y-3)$$

Tentukan nilai dari f(20,10) dan buat grafiknya di f(20,10:30,20)!

3. Sketsakan grafik fungsi dibawah ini pada[-pi,2pi]

$$y(t) = sin(5t)$$

4. Diketahui suatu fungsi f(x):

$$f(x) = 3^x$$

Tentukan nilai f(a+2ˆb-c) jika nilai a=1, b=2, dan c=5!

5. Untuk fungsi:

$$f(x) = 2x^2 - 3x + 20$$

$$g(x) = 3x - 5$$

cari nilai

$$fog(5), gof(8)$$

Definisi limit

Pada dasarnya, limit digunakan untuk menyatakan sesuatu yang nilainya mendekati nilai tertentu. Limit dapat diartikan sebagai menuju suatu batas, sesuatu yang dekat namun tidak dapat dicapai. Mengapa nilainya hanya mendekati? Karena suatu fungsi biasanya tidak terdefinisi pada titik-titik tertentu. Walaupun suatu fungsi seringkali tidak terdefinisi untuk titik tertentu, namun masih dapat dicari tahu berapa nilai yang didekati oleh fungsi tersebut apabila titik tertentu semakin didekati yaitu dengan limit.

Bentuk umum dari limit sendiri dinotasikan dengan:

maxima: 'limit(F(x),x,c)=L

Notasi tersebut menyatakan bahwa f(x) untuk niai x mendekati c sama dengan L. F(x) disini dapat berupa bermacam-macam jenis fungsi. Dan L dapat berupa konstanta, ataupun "und" (tak terdefinisi), "ind" (tak tentu namun terbatas), "infinity" (kompleks tak hingga). Begitupun dengan batas c, dapat berupa sebarang nilai atau pada tak hingga (-inf, minf, dan inf).

Sebuah fungsi dapat dikatakan memiliki limit apabila limit kanan dan limit kiri nya memiliki nilai yang sama. Dimana, limit dari fungsi tersebut adalah nilai dari limit kanan dan limit kiri fungsi yang bernilai sama tadi.

Sifat-sifat Limit Jika f(x) dan g(x) adalah fungsi yang memiliki

limit di c, n merupakan bilangan bulat positif, dan k adalah konstanta.
1.

$$\lim_{x\to c} k = k$$

```
>$showev('limit(5,x,2))
```

$$5 = 5$$

$$\lim_{x\to 2} 5 = 5$$

2.

$$\lim_{x\to c} x = c$$

```
>$showev('limit(x,x,3))
```

$$\lim_{x\to 3} x = 3$$

3.

$$\lim_{x\to c} kf\left(x\right)=k\,\left(\lim_{x\to c} 8^{x}+27\right)$$

```
>$showev('limit(4*x^2,x,1))
```

$$4\,\left(\lim_{x\to 1} x^2\right)=4$$

4.

$$\lim_{x\to c} \left[8^{x}+3\,x+28\right]=\lim_{x\to c} 8^{x}+27+\lim_{x\to c} 3\,x+1$$

```
>$showev('limit(x^3+2*x,x,1))
```

$$\lim_{x\to 1} x^3+2\,x=3$$

5.

$$\lim_{x\to c}\left[-8^x+3\,x-26\right]=\lim_{x\to c}3\,x+1-\lim_{x\to c}8^x+27$$

```
>$showev('limit(x^2-4*x,x,5))
```

$$\lim_{x\to 5}x^2-4\,x=5$$

6.

$$\lim_{x\to c}\left[\left(3\,x+1\right)\,\left(8^x+27\right)\right]=\left(\lim_{x\to c}3\,x+1\right)\,\left(\lim_{x\to c}8^x+27\right)$$

```
>$showev('limit((x-1)*(x+4),x,3))
```

$$\lim_{x\to 3}\left(x-1\right)\,\left(x+4\right)=14$$

7.

$$\lim_{x\to c} \frac{3\,x+1}{8^x+27} = \frac{\lim_{x\to c} 3\,x+1}{\lim_{x\to c} 8^x+27}$$

```
>$showev('limit((x+5)/(x+1),x,0))
```

$$\lim_{x\to 0} \frac{x+5}{x+1} = 5$$

8.

$$\lim_{x\to c} \left(8^x+27\right)^n = \left(\lim_{x\to c} 8^x+27\right)^n$$

```
>$showev('limit((x+3)^2,x,1))
```

$$\lim_{x\to 1} \left(x+3\right)^2 = 16$$

9.

$$\lim_{x\to c}\left(8^{x}+27\right)^{\frac{1}{n}}=\left(\lim_{x\to c}8^{x}+27\right)^{\frac{1}{n}}$$

```
>$showev('limit((x^2+3*x+1)^(1/2),x,2))
```

$$\lim_{x\rightarrow 2}{\sqrt{x^2+3\,x+1}}=\sqrt{11}$$

Limit pada EMT

Pada EMT cara mendefinisikan limit yaitu dengan format :

$showev('limit(f(x),x,c))

Format tersebut akan menampilkan limit yang dimaksud dan hasilnya. Jika kita ingin menampilkan hasilnya saja dari sebuah limit tanpa menampilkan limitnya, kita bisa menggunakan format :

$limit(f(x),x,c)

Sedangkan, untuk limit kanan dan limit kiri seperti pada definisi dapat ditampilkan di EMT dengan cara menambah opsi "plus" atau "minus" :

$showev('limit(f(x),x,c, plus)) atau 'limit(f(x),x,c, minus)

```
>$showev('limit(x^2+3*x+4,x,2))
```

$$\lim_{x\rightarrow 2}{x^2+3\,x+4}=14$$

```
>$limit(x^2+3*x+4,x,2)
```


Menghitung dan Visualisasi 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).

Limit dapat divisualisasikan menggunakan plot 2 dimensi. Pada EMT sendiri, format yang bisa digunakan untuk memvisualisasikan limit adalah :

plot2d("f(x)",-c,c):

Dengan f(x) adalah fungsi pada limit yang dicari, dan c berupa bilangan real menyesuaikan batas dari limit itu sendiri.

Limit Aljabar

1. Tunjukkan bahwa limit kiri dan kanan dari fungsi berikut bernilai sama. Berapakah nilai limitnya? maxima: 'limit(x^2+x-1,x,0)

```
>$showev('limit(x^2+x-1,x,0,minus))
```

$$\lim_{x\uparrow 0} x^2 + x - 1 = -1$$

```
>$showev('limit(x^2+x-1,x,0,plus))
```

$$\lim_{x\downarrow 0} x^2+x-1=-1$$

Dapat terlihat bahwa nilai limit kiri = nilai limit kanan. Maka fungsi tersebut memiliki nilai limit, yaitu

```
>$showev('limit(x^2+x-1,x,0))
```

$$\lim_{x\to 0} x^2+x-1=-1$$

2. Tunjukkan bahwa limit kiri dan kanan dari fungsi berikut bernilai sama. Berapakah nilai limitnya? maxima: 'limit(sqrt(xˆ2+x-1),x,3)

```
>$showev('limit(sqrt(x^2+x-1),x,3,minus))
```

$$\lim_{x\uparrow 3} \sqrt{x^2+x-1}=\sqrt{11}$$

```
>$showev('limit(sqrt(x^2+x-1),x,3,plus))
```

$$\lim_{x\downarrow 3}\sqrt{x^2+x-1}=\sqrt{11}$$

Dapat terlihat bahwa nilai limit kiri = nilai limit kanan. Maka fungsi tersebut memiliki nilai limit, yaitu

```
>$showev('limit(sqrt(x^2+x-1),x,3))
```

$$\lim_{x\rightarrow 3}\sqrt{x^2+x-1}=\sqrt{11}$$

3. Tunjukkan bahwa limit kiri dan kanan dari fungsi berikut bernilai sama. Berapakah nilai limitnya?

$$\lim_{x\rightarrow -3}\frac{|x+3|}{x+3}$$

```
>$showev('limit(abs(x+3)/(x+3),x,-3,minus))
```

$$\lim_{x\uparrow -3}\frac{|x+3|}{x+3}=-1$$

```
>$showev('limit(abs(x+3)/(x+3),x,-3,plus))
```

$$\lim_{x\downarrow -3}{\frac{\left| x+3\right| }{x+3}}=1$$

Karena nilai limit kiri tidak sama dengan nilai limit kanan. Maka fungsi diatas tidak memiliki limit di x=-3

```
>$showev('limit(abs(x+3)/(x+3),x,-3))
```

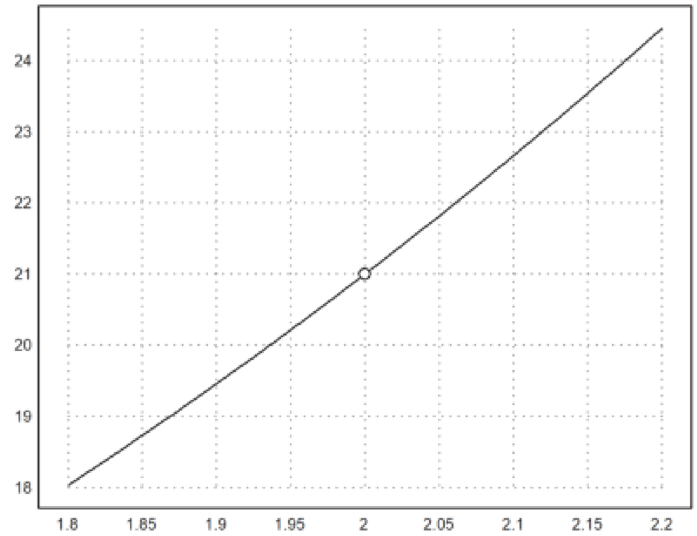
$$\lim_{x\rightarrow -3}{\frac{\left| x+3\right| }{x+3}}={\it und}$$

4. Berapakah nilai limit dari fungsi berikut? Tunjukkan dengan menggunakan grafik!

$$\lim_{x\rightarrow 2}{x^3+4\,x+5}$$

```
>$limit(x^3+4*x+5,x,2)
```

```
>plot2d("x^3+4*x+5",1.8,2.2); plot2d(2,21,>points,style="ow",>add):
```

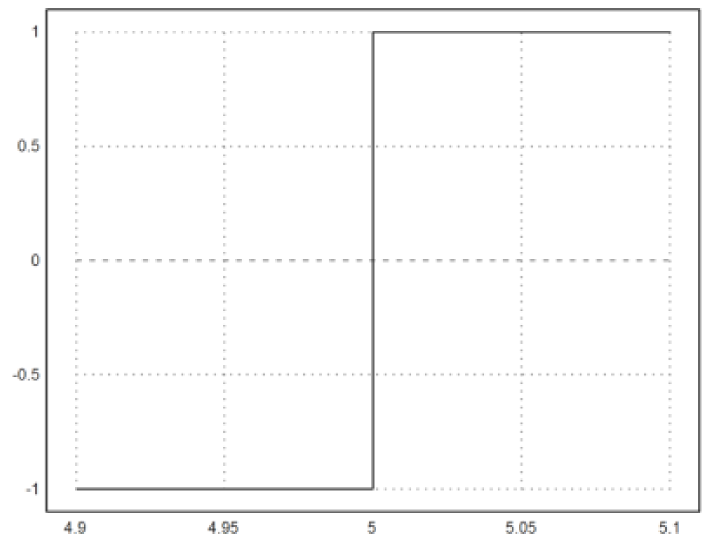


Jadi berdasarkan grafik, nilai limit tersebut adalah 21.

5.Berapakah nilai limit dari fungsi berikut? Tunjukkan dengan menggunakan grafik!

$$\lim_{x\to 5} \frac{|x-5|}{x-5}$$

```
>plot2d("abs(x-5)/(x-5)",4.9,5.1):
```



Karena nilai limit kiri tidak sama dengan limit kanan. maka nili limit fungsi tersebut tidak ada.

6. Berapakah nilai limit dari fungsi berikut?

$$\lim_{x\to\infty} \frac{x^5 - 3\,x^3 + x^2 - 9}{4\,x^5 - 7\,x^4 + 2\,x - 1}$$

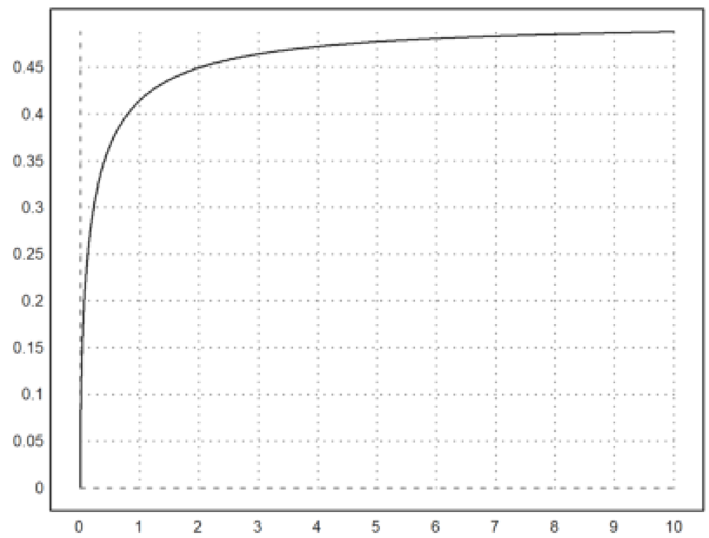
```
>$showev('limit((x^5-3*x^3+x^2-9)/(4*x^5-7*x^4+2*x-1),x,inf))
```

$$\lim_{x\to\infty} \frac{x^5 - 3\,x^3 + x^2 - 9}{4\,x^5 - 7\,x^4 + 2\,x - 1} = \frac{1}{4}$$

7. Tunjukkan nilai limit fungsi berikut dengan grafik!

$$\lim_{x\to\infty} \sqrt{x^2+x} - x$$

```
>plot2d("sqrt(x^2+x)-x",0,10):
```



Dengan menggunakan grafik diperlihatkan bahwa nilai limit mendekati 0.5. Mari kita buktikan di baris perintah

```
>$showev('limit(sqrt(x^2+x)-x,x,inf))
```

$$\lim_{x\to\infty}\sqrt{x^2+x}-x=\frac{1}{2}$$

Limit Trigonometri

1. Tunjukkan bahwa limit kiri dan kanan dari fungsi berikut bernilai sama. Berapakah nilai limitnya?

$$\lim_{x\to 0}\cos x\,\sin x$$

```
>$showev('limit(cos(x)*sin(x),x,0,minus))
```

$$\lim_{x\uparrow 0}\cos x\,\sin x=0$$

```
>$showev('limit(cos(x)*sin(x),x,0,plus))
```

$$\lim_{x\downarrow 0}\cos x\,\sin x=0$$

Dapat terlihat bahwa nilai limit kiri = nilai limit kanan. Maka fungsi tersebut memiliki nilai limit, yaitu

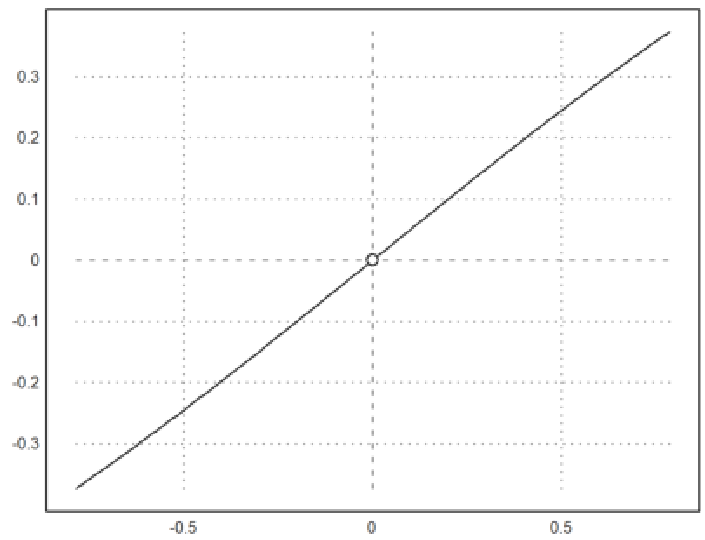
```
>$showev('limit(cos(x)*sin(x),x,0))
```

$$\lim_{x\to 0}\cos x\,\sin x=0$$

2. Berapakah nilai limit dari fungsi berikut? Tunjukkan dengan menggunakan grafik

$$\lim_{x\to 0}\frac{1-\cos x}{x}$$

```
>plot2d("(1-cos(x))/x",-pi/4,pi/4); plot2d(0,0,>points,style="ow",>add):
```



3. Tunjukkan bahwa limit kiri dan kanan dari fungsi berikut bernilai sama. Berapakah nilai limitnya?

$$\lim_{x \to 0} \frac{\sin x}{1 - \cos x}$$

```
>$showev('limit(3*sin(x)^2/(1-cos(x)),x,0,minus))
```

$$3 \left(\lim_{x \uparrow 0} \frac{\sin^2 x}{1 - \cos x} \right) = 6$$

```
>$showev('limit(3*sin(x)^2/(1-cos(x)),x,0,plus))
```

$$3\left(\lim_{x\downarrow 0}\frac{\sin^2 x}{1-\cos x}\right)=6$$

Dapat terlihat bahwa nilai limit kiri = nilai limit kanan. Maka fungsi tersebut memiliki nilai limit, yaitu

```
>$showev('limit(3*sin(x)^2/(1-cos(x)),x,0))
```

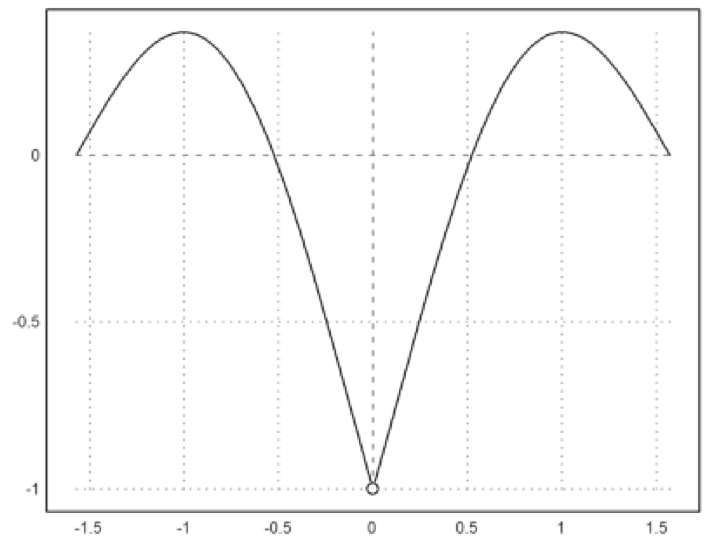
$$3\left(\lim_{x\to 0}\frac{\sin^2 x}{1-\cos x}\right)=6$$

4. Berapakah nilai limit dari fungsi berikut? Tunjukkan dengan menggunakan grafik!
maxima: 'limit(abs(sin(2*x))-cos(x),x,0)

```
>$limit(abs(sin(2*x))-cos(x),x,0)
```

$$-1$$

```
>plot2d("abs(sin(2x))-cos(x)",-pi/2,pi/2); plot2d(0,-1,>points,style="ow",>add):
```



Jadi berdasarkan grafik, nilai limit tersebut adalah -1.

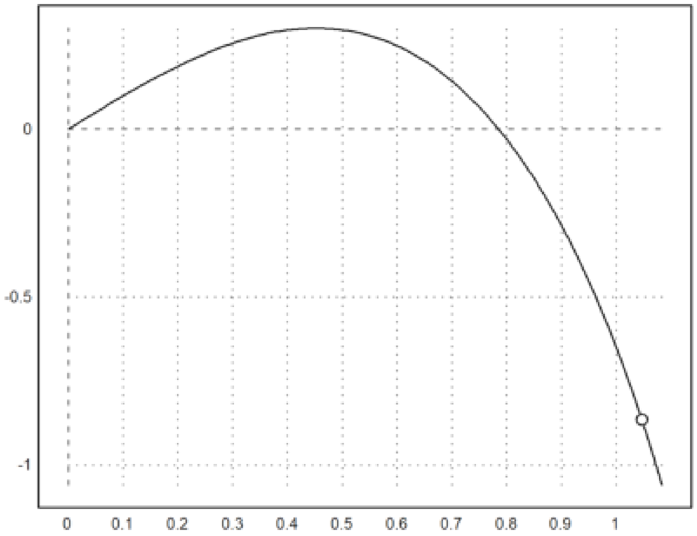
5. Berapakah nilai limit dari fungsi berikut? Tunjukkan dengan menggunakan grafik!

$$\lim_{x \to \frac{\pi}{3}} 2\cos x\,\sin x - \tan x$$

```
>$limit(2*cos(x)*sin(x)-tan(x),x,pi/3)
```

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

```
>plot2d("2*cos(x)*sin(x)-tan(x)",0,pi/2.9); plot2d(pi/3,-(3)^(1/2)/2,>points,style="ow",>add):
```



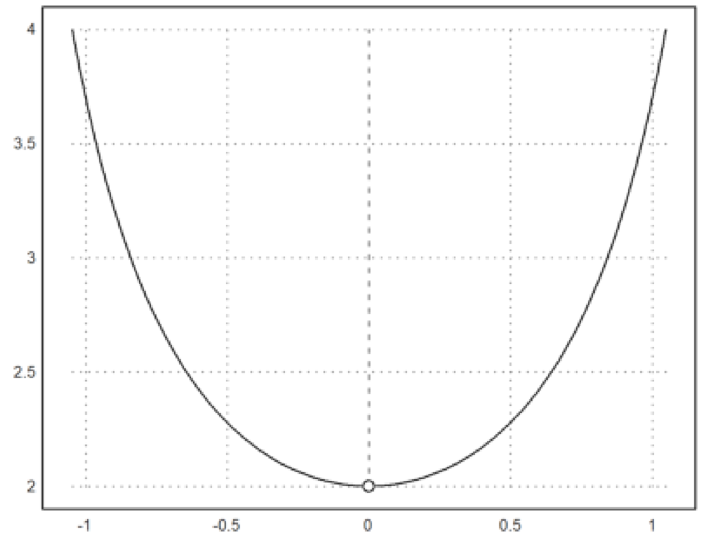
Jadi berdasarkan grafik, nilai limit tersebut adalah -0.866.

6. Berapakah nilai limit dari fungsi berikut? Tunjukkan dengan menggunakan grafik!
maxima: 'limit(2*sec(x),x,0)

```
>$limit(2*sec(x),x,0)
```

2

```
>plot2d("2*sec(x)",-pi/3,pi/3); plot2d(0,2,>points,style="ow",>add):
```



Jadi berdasarkan grafik, nilai limit tersebut adalah 2.

Latihan

Tunjukkan bahwa limit kiri dan kanan dari fungsi berikut bernilai sama.

$$\lim_{x\to 3}\frac{(x-3)\,\left|x^2+5\,x+8\right|}{\cos x}$$

```
>$limit(abs(x^2+5*x+8)/cos(x)*(x-3),x,3)
```

$$0$$

3. Turunan Fungsi

`Turunan adalah pengukuran terhadap bagaimana fungsi berubah seiring`

perubahan nilai yang dimasukkan, atau secara umum turunan menunjukkan bagaimana suatu besaran berubah akibat perubahan besaran lainnya. Proses dalam menemukan turunan disebut diferensiasi.

Definisi turunan:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Berikut adalah contoh-contoh menentukan turunan fungsi dengan beberapa cara.

Turunan fungsi Aljabar

Mencari turunan dari

$$f(x) = x^n$$

```
>$showev('limit(((x+h)^n-x^n)/h,h,0)) // turunan x^n
```

$$\lim_{h\to 0} \frac{(x+h)^n - x^n}{h} = n\, x^{n-1}$$

Bukti:

$$(x+h)^n = \binom{n}{0} x^{n-0} h^0 + \binom{n}{1} x^{n-1} h^1 + \binom{n}{2} x^{n-2} h^2 + ... + \binom{n}{n} x^{n-n} h^n$$

$$(x+h)^n = x^n + nx^{n-1}h + \binom{n}{2} x^{n-2} h^2 + ... + h^n$$

$$(x+h)^n - x^n = nx^{n-1}h + \binom{n}{2} x^{n-2} h^2 + ... + h^n$$

$$\frac{(x+h)^n - x^n}{h} = nx^{n-1} + \binom{n}{2} x^{n-2} h + ... + h^{n-1}$$

$$\lim_{h\to 0} \frac{(x+h)^n - x^n}{h} = nx^{n-1}$$

Mencari turunan dari

$$f(x) = x^2$$

```
>$showev('limit(((x+h)^2-x^2)/h,h,0)) // turunan x^2
```

$$\lim_{h\to 0}{\frac{\left(x+h\right)^2-x^2}{h}}=2\,x$$

```
>p &= expand((x+h)^2-x^2)|simplify; $p //pembilang dijabarkan dan disederhanakan
```

$$2\,h\,x+h^2$$

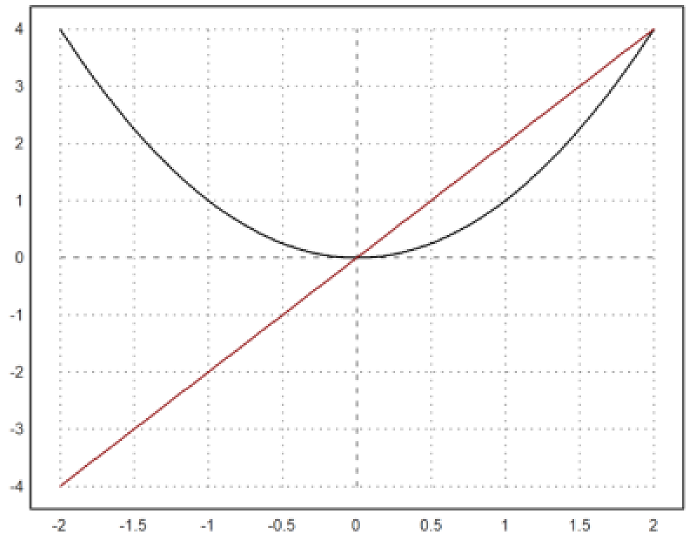
```
>q &=ratsimp(p/h); $q // ekspresi yang akan dihitung limitnya disederhanakan
```

$$2\,x+h$$

```
>$limit(q,h,0) // nilai limit sebagai turunan
```

$$2\,x$$

```
>plot2d(["x^2", "2x"], color=[black,red]):
```



Mencari turunan dari

$$f(x) = f(x)^x$$

```
>$showev('limit((f(x+h)-f(x))/h,h,0)) // turunan f(x)=f(x)^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((f(x+h)-f(x))/h,h,0)) // turunan f(x)=f(x)^x ...
                                     ^
```

Di sini Maxima 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)=f(x)^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:
... ssume(x>0); $showev('limit((f(x+h)-f(x))/h,h,0)) // turunan f( ...
                                                      ^
```

```
>&forget(x>0) // jangan lupa, lupakan asumsi untuk kembali ke semula
```

```
[x > 0]
```

```
>&forget(x<0)
```

```
[x < 0]
```

```
>&facts()
```

```
[kind(sinh, one_to_one), kind(log, one_to_one),
                kind(tanh, one_to_one), kind(log, increasing)]
```

Selain dengan showev('limit...) kita juga dapat mencari turunan dengan menyederhanakan pembilang seperti di awal

```
>p &= expand(f(x+h)-f(x))|simplify;  $p
```

$$8^{x+h}-8^x$$

```
>q &=ratsimp(p/h); $q
```

$$\frac{\left(8^h-1\right)\,8^x}{h}$$

```
>$limit(q,h,0)
```

```
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:
 $limit(q,h,0) ...
             ^
```

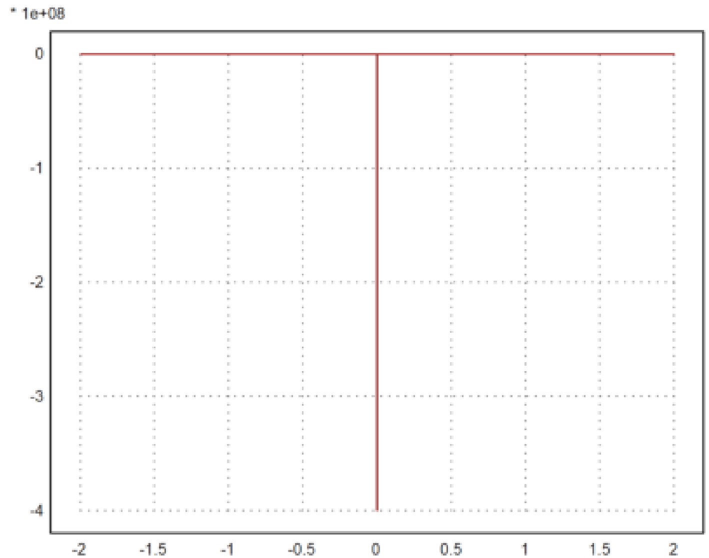
Mencari turunan dari

$$f(x) = \frac{1}{x}$$

```
>$showev('limit((1/(x+h)-1/x)/h,h,0)) // turunan 1/x
```

$$\lim_{h\rightarrow 0}{\frac{\frac{1}{x+h}-\frac{1}{x}}{h}}=-\frac{1}{x^2}$$

```
>plot2d("-x^(-2)", color=red):
```



Turunan fungsi trigonometri

Mencari turunan dari

$$sin(x)$$

```
>$showev('limit((sin(x+h)-sin(x))/h,h,0)) // turunan sin(x)
```

$$\lim_{h\to 0}{\frac{\sin \left(x+h\right)-\sin x}{h}}=\cos x$$

Bukti:

$$sin(x+h) = sin(x)cos(h) + cos(x)sin(h)$$

$$sin(x+h) - sin(x) = sin(x)cos(h) + cos(x)sin(h) - sin(x)$$

$$sin(x+h) - sin(x) = cos(x)sin(h) - sin(x) + sin(x)cos(h)$$

$$sin(x+h) - sin(x) = cos(x)sin(h) - sin(x)(1 - cos(h))$$

$$\lim_{h\to 0} \frac{sin(x+h) - sin(x)}{h} = \lim_{h\to 0} \frac{cos(x)sin(h)}{h} - \lim_{h\to 0} \frac{sin(x)(1 - cos(h)}{h}$$

$$\lim_{h\to 0} \frac{sin(x+h) - sin(x)}{h} = cos(x) \lim_{h\to 0} \frac{sin(h)}{h} - sin(x) \lim_{h\to 0} \frac{(1 - cos(h)}{h}$$

$$= cos(x) \times 1 - sin(x) \times 0 = cos(x)$$

```
>p &= expand((sin(x+h)-sin(x)))|simplify; $p
```

$$\sin(x+h) - \sin x$$

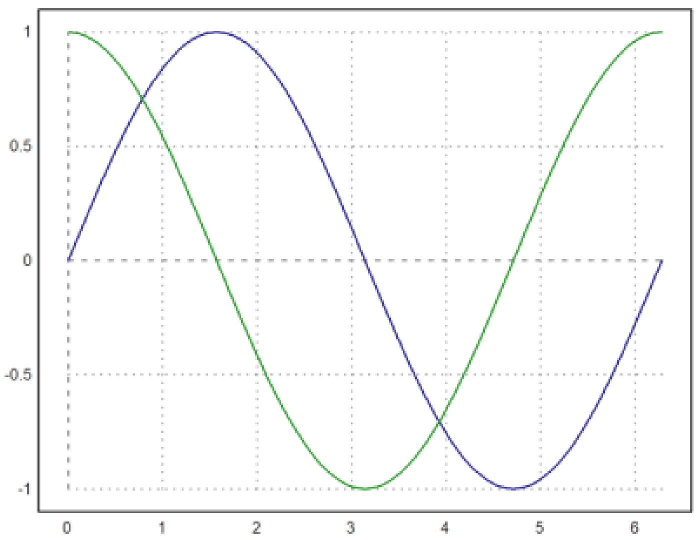
```
>q &=ratsimp(p/h); $q
```

$$\frac{\sin(x+h) - \sin x}{h}$$

```
>$limit(q,h,0) // nilai limit sebagai turunan
```

$$\cos x$$

```
>plot2d(["sin(x)", "cos(x)"],0, 2*pi, color=[blue,green]):
```



Mencari turunan dari

$$tan(x)$$

```
>$showev('limit((tan(x+h)-tan(x))/h,h,0)) // turunan tan(x)
```

$$\lim_{h\to 0}\frac{\tan\left(x+h\right)-\tan x}{h}=\frac{1}{\cos^2 x}$$

Bukti:

$$tan(x+h) = \frac{tan(x) + tan(h)}{1 - tan(x)tan(h)}$$

$$tan(x+h) - tan(x) = \frac{tan(x) + tan(h) - tan(x) + tan^2(x)tan(h)}{1 - tan(x)tan(h)}$$

$$\frac{tan(x+h) - tan(x)}{h} = \frac{\frac{tan(x)+tan(h)-tan(x)+tan^2(x)tan(h)}{1-tan(x)tan(h)}}{h}$$

$$= \frac{tan(h) + tan^2(x)tan(h)}{h(1 - tan(x)tan(h)}$$

$$= \lim_{h\to 0} \frac{tan(h) + tan^2(x)tan(h)}{h(1 - tan(x)tan(h)}$$

$$= \lim_{h\to 0} \frac{1 + tan^2(x)}{1 - tan(x)tan(h)} \cdot \lim_{h\to 0} \frac{tan(h)}{h}$$

$$= \frac{1 + tan^2(x)}{1 - tan(x)tan(0)} \cdot 1$$

$$= 1 + tan^2(x)$$

$$1 + tan^2(x) = 1 + \frac{sin^2(x)}{cos^2(x)}$$

$$= \frac{cos^2(x) + sin^2(x)}{cos^2(x)}$$

$$= \frac{1}{cos^2(x)}$$

```
>p &= expand((tan(x+h)-tan(x)))|simplify; $p
```

$$\tan \left(x+h\right)-\tan x$$

```
>q &=ratsimp(p/h); $q
```

$$\frac{\tan \left(x+h\right)-\tan x}{h}$$

```
>$limit(q,h,0) // nilai limit sebagai turunan
```

$$\frac{1}{\cos ^2x}$$

Mencari turunan dari

$$arcsin(x)$$

```
>$showev('limit((asin(x+h)-asin(x))/h,h,0)) // turunan arcsin(x)
```

$$\lim_{h\rightarrow 0}{\frac{\arcsin \left(x+h\right)-\arcsin x}{h}}=\frac{1}{\sqrt{1-x^2}}$$

Bukti:

$$f'(x) = \lim_{h \to 0} \frac{arcsin(x+h) - arcsin(x)}{h}$$

Misalkan arcsin(x+h)=A dan arcsin(x)=B

Sehingga sin A= x+h dan sin B=x
Kurangkan persamaan kedua dengan persamaan pertama

$$sin(A) - sin(B) = (x+h) - x$$

$$sin(A) - sin(B) = h$$

karena

$$h \to 0, sin(A) - sin(B) \to 0$$

maka

$$sin(A) \to sin(B), A \to B, A - B \to 0$$

sehingga f'(x) menjadi

$$f'(x) = \lim_{A-B \to 0} \frac{A - B}{sinA - sinB}$$

Identitas trigonometri

$$sin(A) - sin(B) = 2sin[\frac{A-B}{2}]cos[\frac{A+B}{2}]$$

$$f'(x) = \lim_{A-B\to 0} \frac{A-B}{2sin[\frac{A-B}{2}]cos[\frac{A+B}{2}]}$$
$$f'(x) = \lim_{A-B\to 0} \frac{\frac{2(A-B)}{2}}{2sin[\frac{A-B}{2}]cos[\frac{A+B}{2}]}$$

ingat bahwa

$$\lim_{x\to 0} \frac{x}{sin(x)} = 1$$

$$f'(x) = \lim_{A-B\to 0} \frac{\frac{(A-B)}{2}}{sin[\frac{A-B}{2}]} \cdot \frac{2}{2cos[\frac{A+B}{2}]}$$
$$f'(x) = 1 \cdot \frac{2}{2cos[\frac{B+B}{2}]}$$
$$f'(x) = \frac{1}{cos(B)}$$

$$f'(x) = \frac{1}{\sqrt{(cos^2(B)}}$$

$$f'(x) = \frac{1}{\sqrt{(1 - sin^2(B)}} = \frac{1}{\sqrt{(1 - x^2}}$$

```
>p &= expand((asin(x+h)-asin(x)))|simplify; $p //pembilang dijabarkan dan disederhanakan
```

$$\arcsin\left(x+h\right)-\arcsin x$$

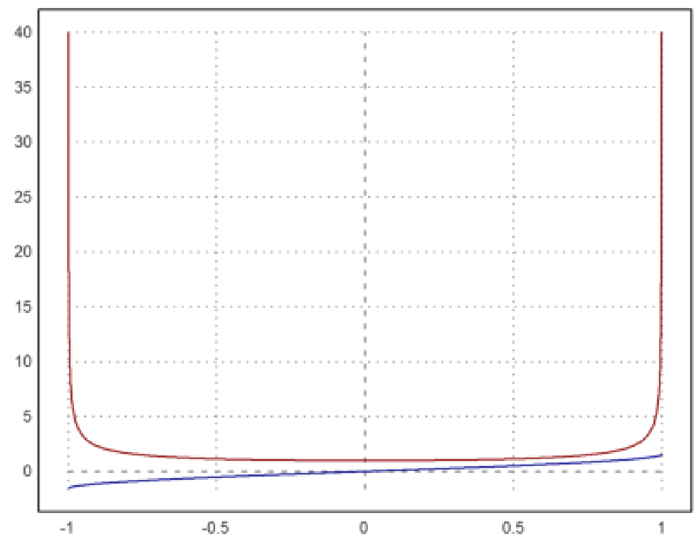
```
>q &=ratsimp(p/h); $q // ekspresi yang akan dihitung limitnya disederhanakan
```

$$\frac{\arcsin\left(x+h\right)-\arcsin x}{h}$$

```
>$limit(q,h,0) // nilai limit sebagai turunan
```

$$\frac{1}{\sqrt{1-x^2}}$$

```
>plot2d(["asin(x)","1/(sqrt(1-x^2))"],-1,1,color=[blue,red]):
```



Mencari turunan dari

$$x^2 + 10$$

di titik x=5

```
>function f(x) &= x^2+10 // definisikan f(x)=x^2+10
```

```
                 2
                x  + 10
```

```
>function df(x) &= showev('diff(f(x),x)); $df(x)
```

$$\frac{d}{d\,x}\,\left(x^2+10\right)=2\,x$$

```
>$% with x=5
```

$$\%at\left(\frac{d}{d\,x}\,\left(x^2+10\right),x=5\right)=10$$

```
>diff(f,5), diffc(f,5)
```

```
9.99999999997
10
```

diff: Diferensiasi numerik yang pada dasarnya kurang akurat untuk fungsi umum.

diffc: Menghitung turunan pertama untuk fungsi analitik nyata dengan menggunakan bagian imajiner dari f(x+ih)/h. Cara ini memungkinkan untuk mengevaluasi f dengan lebih akurat.

Mencari turunan dari

$$sinh(x)$$

```
>function f(x) &= sinh(x) // definisikan f(x)=sinh(x)
```

```
sinh(x)
```

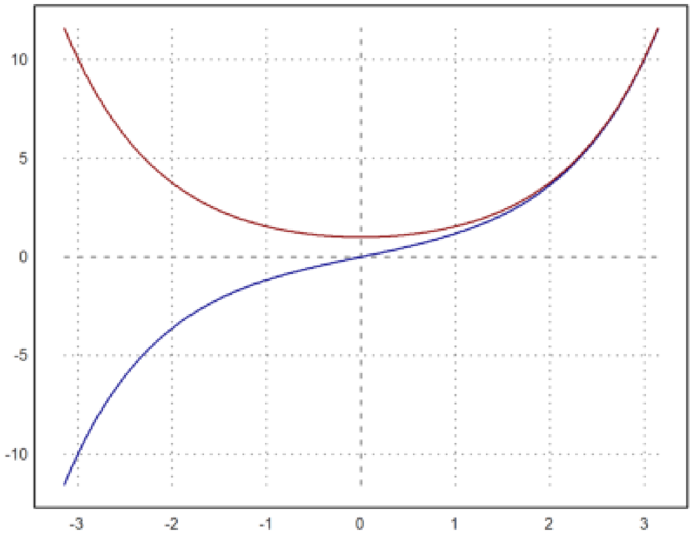
```
>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x) // df(x) = f'(x)
```

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

Hasilnya adalah cosh(x), karena

$$\frac{e^x+e^{-x}}{2}=\cosh(x).$$

```
>plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]):
```




```
>$showev('diff(f(x),x))
```

$$\frac{d}{d\,x}\,\sinh x = \cosh x$$

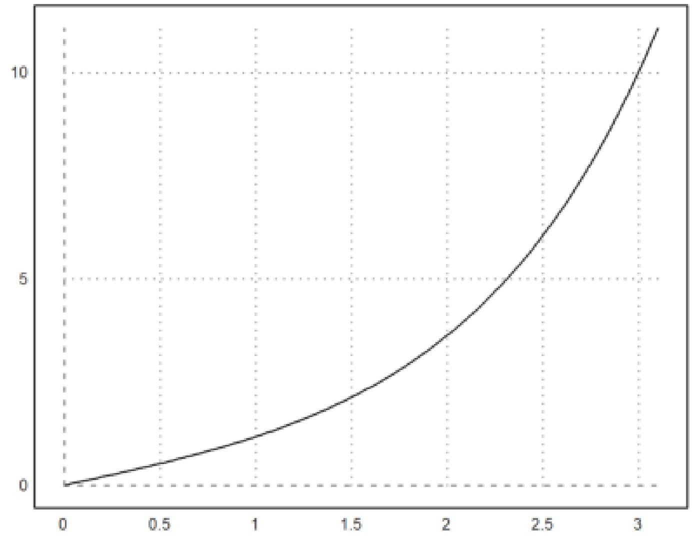
```
>$% with x=3
```

$$\%at\left(\frac{d}{d\,x}\,\sinh x, x = 3\right) = \cosh 3$$

```
>$float(%)
```

$$\%at\left(\frac{d^{1.0}}{d\,x^{1.0}}\,\sinh x, x=3.0\right)=10.06766199577777$$

```
>plot2d(f,0,3.1):
```



Mencari turunan dari

$$f(x) = sin(3x^5 + 7)^2$$

```
>function f(x) &= sin(3*x^5+7)^2
```

```
                          2      5
                       sin (3 x  + 7)
```

```
>$showev('diff(f(x),x))
```

$$\frac{d}{d\,x}\,\sin^2\left(3\,x^5+7\right)=30\,x^4\,\cos\left(3\,x^5+7\right)\,\sin\left(3\,x^5+7\right)$$

Kita akan mencari nilai turunan tersebut ketika x=2

```
>$% with x = 2
```

$$\%at\left(\frac{d}{d\,x}\,\sin^2\left(3\,x^5+7\right) , x=2\right)=480\,\cos 103\,\sin 103$$

```
>$float(%)
```

$$\%at\left(\frac{d^{1.0}}{d\,x^{1.0}}\,\sin^2\left(3.0\,x^5+7.0\right),x=2.0\right)=-233.9140567500984$$

```
>diff(f,2), diffc(f,2)
```

```
Variable or function f not found.
Error in:
diff(f,2), diffc(f,2) ...
      ^
```

diff: Diferensiasi numerik yang pada dasarnya kurang akurat untuk fungsi umum.

diffc: Menghitung turunan pertama untuk fungsi analitik nyata dengan menggunakan bagian imajiner dari f(x+ih)/h. Cara ini memungkinkan untuk mengevaluasi f dengan lebih akurat.

```
>plot2d(f,0,3.1) :
```

```
Variable or function f not found.
Error in:
plot2d(f,0,3.1) : ...
        ^
```

Mencari turunan dari

$$a(x) = 5cos(2x) - 2xsin(2x)$$

```
>function a(x) &=5*cos(2*x)-2*x*sin(2*x) // mendifinisikan fungsi a
```

```
5 cos(2 x) - 2 x sin(2 x)
```

```
>function da(x) &=diff(a(x),x); $da(x) // da(x) = a'(x)
```

$$-12\,\sin\left(2\,x\right)-4\,x\,\cos\left(2\,x\right)$$

```
>$'a(1)=a(1), $float(a(1)), $'a(2)=a(2), $float(a(2)) // nilai a(1) dan a(2)
```

$$-3.899329036387075$$

$$-0.2410081230863468$$

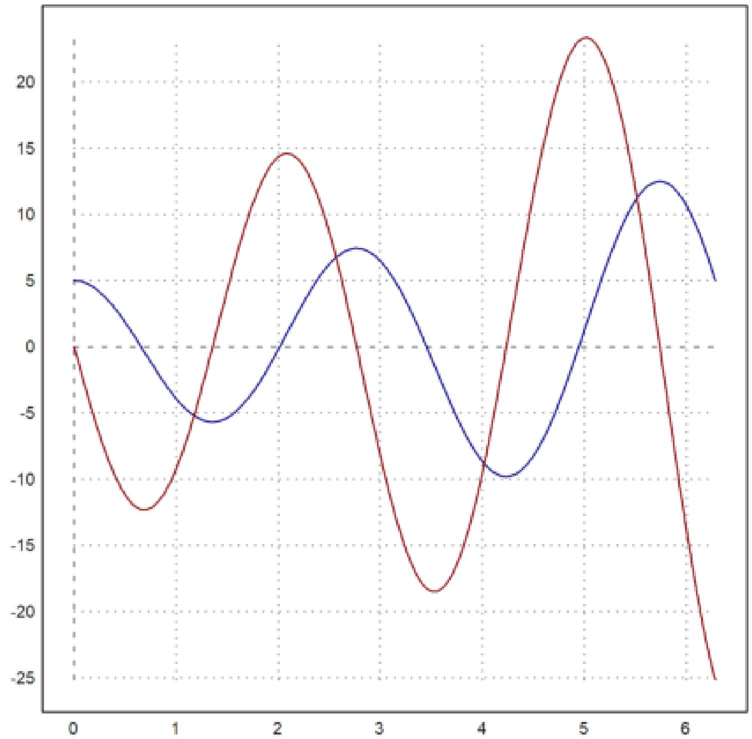
```
>xp=solve("da(x)",1,2,0) // solusi a'(x)=0 pada interval [1, 2]
```

```
1.35822987384
```

```
>da(xp), a(xp) // cek bahwa a'(xp)=0 dan nilai ekstrim di titik tersebut
```

```
0
-5.67530133759
```

```
>plot2d(["a(x)","da(x)"],0,2*pi,color=[blue,red]): //grafik fungsi dan turunannya
```



Pada titik puncak grafik a(x), nilai turunan a'(x) akan selalu sama dengan nol. Ini karena gradien garis singgung pada titik puncak adalah nol.

Turunan fungsi logaritma

Mencari turunan dari

$$f(x) = log(x)$$

```
>$showev('limit((log(x+h)- log(x))/h,h,0)) // turunan log(x)
```

$$\lim_{h\to 0}\frac{\log\left(x+h\right)-\log x}{h}=\frac{1}{x}$$

Bukti:

$$f'(x) = \lim_{h\to 0}\frac{log(x+h) - logx}{h}$$

ingat bahwa

$$log(a) - log(b) = log(\frac{a}{b})$$

$$f'(x) = \lim_{h\to 0}\frac{log(\frac{x+h}{x})}{h}$$

$$f'(x) = \lim_{h\to 0}\frac{log(1+\frac{h}{x})}{h}$$

misalkan h/x = n sehingga h=nx

$$f'(x) = \lim_{n \to 0} \frac{log(1+n)}{nx}$$
$$f'(x) = \lim_{n \to 0} \frac{1}{nx} \cdot log(1+n)$$

ingat bahwa

$$alogb = logb^a$$

$$f'(x) = \lim_{n \to 0} \frac{1}{x} \cdot log(1+n)^{\frac{1}{n}}$$
$$f'(x) = \frac{1}{x} log \lim_{n \to 0} (1+n)^{\frac{1}{n}}$$

$$\lim_{n \to 0} (1+n)^{\frac{1}{n}} = e$$

$$f'(x) = \frac{1}{x} loge$$
$$f'(x) = \frac{1}{x}$$

Turunan fungsi eksponensial

Mencari turunan dari

$$f(x) = e^x$$

```
>$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:
                                       ^
```

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('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
```

$$\left(\lim_{h\rightarrow 0}{\frac{e^{h}-1}{h}}\right)\,e^{x}=e^{x}$$

```
>$showev('limit((E^h-1)/h,h,0))
```

$$\lim_{h\rightarrow 0}{\frac{e^{h}-1}{h}}=1$$

Sifat- sifat Turunan

1. Jika f(x)=k dengan k suatu konstanta maka untuk sebarang x, f'x = 0

```
Bukti:
```

$$f'(x) = \lim_{h\to 0} \frac{f(x+h)-f(x)}{h} = \lim_{h\to 0} \frac{k-k}{h} = \lim_{h\to 0} 0 = 0$$

2. Jika f(x)=x, maka f'(x)=1

```
Bukti:
```

$$f'(x) = \lim_{h\to 0} \frac{f(x+h)-f(x)}{h} = \lim_{h\to 0} \frac{x+h-x}{h} = \lim_{h\to 0} \frac{h}{h} = 1$$

3. Jika k suatu konstanta dan f suatu fungsi yang terdiferensialkan,

```
maka (kf)'(x)= kf'(x)
Bukti:
Andaikan F(x) = kf(x), maka
```

$$\begin{aligned} F'(x) &= \lim_{h\to 0} \frac{F(x+h)-F(x)}{h} = \lim_{h\to 0} \frac{kf(x+h)-kf(x)}{h} \\ &= \lim_{h\to 0} k\frac{f(x+h)-f(x)}{h} = k\lim_{h\to 0} \frac{f(x+h)-f(x)}{h} \\ &= kf'(x) \end{aligned}$$

4. Jika f dan g adalah fungsi-fungsi yang terdiferensial,

```
maka (f+g)'(x)=f'(x)+g'(x)
Bukti:
```

$$
\begin{aligned}
F'(x) &= \lim_{h\to 0} \frac{f(x+h)+g(x+h)-f(x)+g(x)}{h} \\
&= \lim_{h\to 0} [\frac{f(x+h)-f(x)}{h} + \frac{g(x+h)-g(x)}{h}] \\
= \lim_{h\to 0} \frac{f(x+h)-f(x)}{h} &+ \lim_{h\to 0} \frac{g(x+h)-g(x)}{h} = f'(x)+g'(x)
\end{aligned}
$$

5. Andaikan f dan g adalah fungsi yang dapat didiferensialkan,

```
maka (fg)'(x)=f'(x)g(x)+ f(x)g'(x)
Bukti:
Andaikan F(x)=f(x)g(x)
```

$$
\begin{aligned}
F'(x) &= \lim_{h\to 0} \frac{F(x+h)-F(x)}{h} \\
&= \lim_{h\to 0} \frac{f(x+h)g(x+h)-f(x)g(x)}{h} \\
&= \lim_{h\to 0} \frac{f(x+h)g(x+h)-f(x+h)g(x)+f(x+h)g(x)-f(x)g(x)}{h} \\
&= \lim_{h\to 0} [f(x+h)\frac{g(x+h)-g(x)+g(x)f(x+h)-f(x)}{h}] \\
= \lim_{h\to 0} f(x+h) &= \lim_{h\to 0} \frac{g(x+h)-g(x)}{h} + g(x) \lim_{h\to 0} \frac{f(x+h)-f(x)}{h} \\
&= f(x)g'(x)+g(x)f'(x)
\end{aligned}
$$

6. Andaikan f dan g adalah fungsi yang dapat didiferensialkan

`dengan g(x)tidak sama dengan 0, maka:`

$$(\frac{f}{g})(x) = \frac{g(x)f'(x) - f(x)g'(x)}{g(x)^2}$$

`Bukti:`
`Andaikan F(x)=f(x)/g(x)`

$$\begin{aligned} F'(x) &= \lim_{h\to 0} \frac{F(x+h) - F(x)}{h} \\ &= \lim_{h\to 0} \frac{\frac{f(x+h)}{g(x+h} - \frac{f(x)}{g(x)}}{h} \\ &= \lim_{h\to 0} \frac{g(x)f(x+h) - f(x)g(x+h)}{h} \frac{1}{g(x)g(x+h)} \\ &= \lim_{h\to 0} [\frac{g(x)f(x+h) - g(x)f(x) + g(x)f(x) - f(x)g(x+h)}{h} \frac{1}{g(x)g(x+h)}] \\ &= \lim_{h\to 0} [g(x)\frac{f(x+h) - f(x)}{h} - f(x)\frac{g(x+h) - g(x)}{h}] \frac{1}{g(x)g(x+h)} \\ &= [g(x)f'(x) - f(x)g'(x)] \frac{1}{g(x)g(x)} \\ &= \frac{g(x)f'(x) - f(x)g'(x)}{g(x)^2} \end{aligned}$$

1. Persamaan gerak suatu partikel dinyatakan dengan rumus

$$s = f(t) = (3t+1)^{\frac{1}{2}}$$

```
( s dalam meter dan t dalam detik).
Kecepatan partikel tersebut pada saat t=8 adalah ... m/detik.
```

```
Penyelesaian:
```

```
>function f(t) &= (3*t+1)^(1/2); $f(t)
```

$$\sqrt{3\,t+1}$$

```
>function df(t) &= diff(f(t),t); $df(t)
```

$$\frac{3}{2\,\sqrt{3\,t+1}}$$

```
>$df(8); fraction %
```

$$\frac{3}{10}$$

Jadi, kecepatan partikel tersebut pada saat t=8 adalah 0,3 m/detik

2. Sebuah pabrik baju memerlukan x meter kain untuk diproduksi yang dinyatakan dengan fungsi:

$$P(x) = \frac{1}{3}x^2 - 12x + 150$$

(dalam juta rupiah).
Berapa biaya produksi minimum yang dikeluarkan oleh pabrik baju

tersebut?

Penyelesaian:

```
>function P(x) &= (1/3)*x^2-12*x+150; $P(x)
```

$$\frac{x^2}{3} - 12\,x + 150$$

```
>function dP(x) &= diff(P(x),x); $dP(x)
```

$$\frac{2\,x}{3} - 12$$

P(x) akan bernilai minimum jika P'(x)=0, maka:

```
>$dP(x)=0
```

$$\frac{2\,x}{3}-12=0$$

```
>$&solve(dP(x),x)
```

$$[x=18]$$

Dengan demikian, biaya produksinya adalah:

```
>P(18)
```

```
42
```

Jadi, biaya produksi minimum yang dikeluarkan oleh pabrik tersebut

adalah 42 juta rupiah.

3. Sebuah peluru ditembakkan ke atas. Jika tinggi h meter setelah t

detik dirumuskan dengan

$$h(t) = 120t - 5t^2$$

maka berapa tinggi maksimum yang dicapai peluru tersebut?

Penyelesaian:
Diketahui:

$$h(t) = 120t - 5t^2$$

Turunan pertama fungsi h adalah

```
>function h(t) &= 120*t-5*t^2; $h(t)
```

$$120\,t - 5\,t^2$$

```
>function dh(t) &= diff(h(t),t); $dh(t)
```

$$120 - 10\,t$$

Nilai t akan maksimum saat

$$h'(t) = 0$$

```
>$dh(t)=0
```

$$120 - 10\,t = 0$$

```
>$&solve(dh(t),t)
```

$$[t = 12]$$

Ketinggian maksimum yang dapat dicapai peluru saat t=12, yaitu

```
>$h(12)
```

$$720$$

Jadi, ketinggian maksimum peluru adalah 720 meter.

1. Cari f'(4) dari fungsi

$$f(x) = \frac{2x-1}{x^2-x}$$

```
>function f(x) &= (2*x -1)/(x^2-x); $f(x)
```

$$\frac{2\,x-1}{x^2-x}$$

```
>function df(x) &= diff(f(x), x); $df(x)
```

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

```
>$df(4)
```

$$-\frac{25}{144}$$

2. Cari turunan dari

$$g(x) = x^2.sin(x)$$

```
>function g(x) &= x^2*sin(x)
```

```
                 2
                x  sin(x)
```

```
>function dg(x) &= diff (g(x), x)
```

```
                                2
                2 x sin(x) + x  cos(x)
```

3. Cari turunan dari

$$(4x - 7)^2(2x + 3)$$

```
>function f(x) &= (4*x-7)^2*(2*x+3); $f(x)
```

$$(2\,x+3)\,(4\,x-7)^2$$

```
>function df(x) &= diff(f(x), x); $df(x)
```

$$2\,(4\,x-7)^2+8\,(2\,x+3)\,(4\,x-7)$$

```
>F &= expand(df(x))|simplify; $F
```

$$96\,x^2-128\,x-70$$

4. Temukan turunan dari

$$f(x)=ln(x^3+3x-4)$$

4. Integral Tak Tentu

EMT dapat digunakan untuk menghitung integral, baik integral tak tentu maupun integral tentu.

Pada notebook ini akan ditunjukkan perhitungan integral tentu dengan menggunakan Teorema Dasar Kalkulus:

Fungsi untuk menentukan integral adalah integrate. Fungsi ini dapat digunakan untuk menentukan, baik integral tentu maupun tak tentu (jika fungsinya memiliki antiderivatif).

CAKUPAN PEMBAHASAN

Definisi
Cara menulis integral pada EMT
Sifat-sifat
Rumus
Kurva

DEFINISI

Integral Tak Tentu adalah bentuk integral yang variabel integrasinya tidak memiliki batas sehingga integrasi dari sebuah fungsi akan menghasilkan banyak kemungkinan dan hanya dinyatakan sebagai penyelesaian umum. Istilah tak tentu berarti bentuk fungsi F(x) memuat konstanta real sebarang.

Misalkan diketahui suatu fungsi F(x) yang merupakan fungsi umum yang memiliki sifat F'(x)=f(x), maka integral tak tentu merupakan himpunan anti turunan F(x) dari f(x) pada interval negatif tak hingga sampai positif tak hingga yang dinotasikan :

$$F(x) = \int f(x)dx + c$$

MENULIS INTEGRAL PADA LATEX

Untuk menulis dengan Latex menggunakan fungsi \int

$$f(x) = \int x^3 dx$$

$$\int 15x^7 dx$$

$$\int x^4 dx$$

$$\int 7x^3 dx$$

$$\int 5x dx$$

1. Sifat Pangkat

`Jika n adalah sembarang bilangan rasional kecuali -1, maka integral`

tak tentu dari xˆn ditulis :

$$\int x^n dx + c = \frac{x^{n+1}}{n+1} + c$$

Contoh:

$$\int x^2 dx + c = \frac{x^{2+1}}{2+1} + c = \frac{x^3}{3} + c$$

2. Penjumlahan dan Pengurangan

$$\int [g(x) + f(x)]dx = \int g(x)dx + \int f(x)dx$$
$$\int [x^2 + 4x]dx = \int x^2 dx + \int 4x dx$$
$$\int [g(x) - f(x)]dx = \int g(x)dx - \int f(x)dx$$
$$\int [x^7 - 3x]dx = \int x^7 dx - \int 3x dx$$

3. Perkalian

Bayangkan f(x) dan g(x) adalah dua fungsi yang bisa kita hitung integral tak tentu (artinya kita mencari antiturunannya), dan anggap k adalah suatu angka tetap (konstanta). Maka aturannya berlaku:

$$\int kf(x)dx = k\int f(x)dx$$

$$\int 4x^2dx = 4\int x^2dx$$

RUMUS INTEGRAL TAK TENTU

Integral tak tentu berarti nilai atau batasannya belum pasti, sehingga ada nilai konstanta(c) di dalamnya. Jadi rumus dasar integral tak tentu adalah :

```
>$showev('integrate(x^n,x)+c)
```

```
Answering "Is n equal to -1?" with "no"
```

$$\int x^n\,dx + c = \frac{x^{n+1}}{n+1} + c$$

```
>$showev('integrate(x^-1,x)+c)
```

$$\int \frac{1}{x}\ dx + c = \log x + c$$

Integral dari fungsi

$$f(x) = x^{-1}$$

menghasilkan fungsi ln karena berasal dari teorema kalkulus

KURVA FUNGSI INTEGRAL/ANTITURUNAN

Kurva fungsi antiturunan adalah kurva yang menggambarkan hubungan antara suatu fungsi dan antiturunannya. Antiturunan, juga dikenal sebagai integral.

Jangan lupa untuk menuliskan terhadap variabel apa suatu fungsi tersebut diintegralkan

1. kurva antiturunan dari fungsi aljabar

$$f(x) = 4x^3dx$$

2. kurva antiderivatif fungsi trigonometri dengan

$$f(x) = cos(x)$$

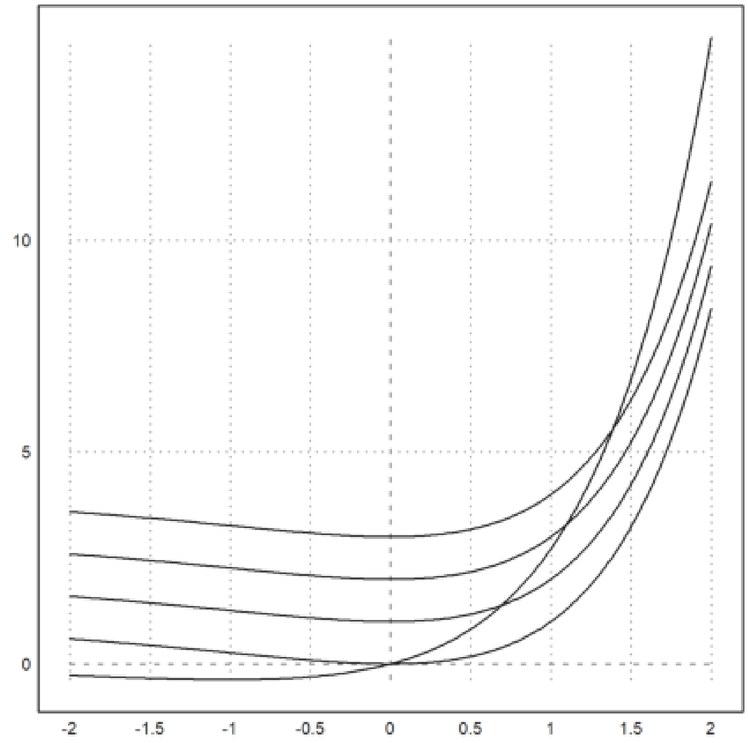
3. kurva antiderivatif dari fungsi eksponensial dengan

$$f(x) = xe^x$$

```
>$showev('integrate(x*E^x,x)+c)
```

$$\int x\, e^x\, dx + c = (x-1)\, e^x + c$$

```
>plot2d(["x*E^x","(x-1)E^x+1","(x-1)E^x+2","(x-1)E^x+3","(x-1)E^x+4"]):
```



4. kurva antiderivatif dari fungsi logaritma dengan

$$f(x) = log(4x)$$

```
>$showev('integrate(log(4*x),x)+c)
```

$$\int \log(4\,x)\;dx + c = \frac{4\,x\;\log(4\,x) - 4\,x}{4} + c$$

Integral Tentu

Pengertian Integral Tentu

Kata “tentu” di sini bermakna sudah pasti atau sudah ditentukan. Oleh karena itu, Integral tentu adalah integral yang sudah ditentukan batasan nilai awal dan akhirnya. Batas dari integral tentu adalah a sampai b atau batas atas sampai batas bawah.

Rumus Integral Tentu

$$\int_a^b f(x)\ dx = F(b) - F(a), \quad \text{dengan } F'(x) = f(x).$$

Contoh:
carilah

$$\int_0^{\pi} sin(x)\ dx$$

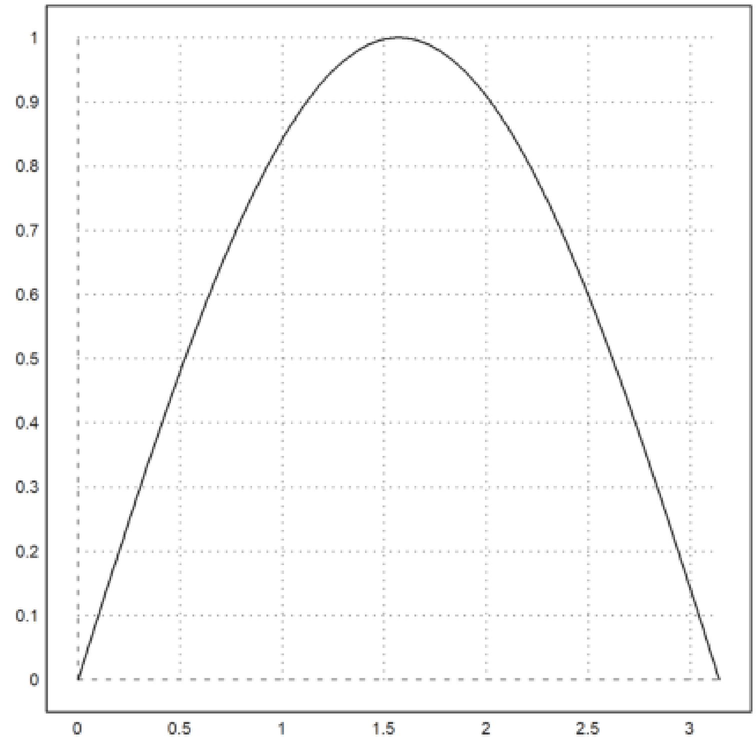
```
>$showev('integrate(sin(x),x,a,b))
```

$$\int_a^b \sin x\ dx = \cos a - \cos b$$

```
>$showev('integrate(sin(x),x,0,pi))
```

$$\int_0^{\pi} \sin x\ dx = 2$$

```
>plot2d("sin(x)",0,pi):
```



Carilah

$$\int_0^2 x^2\sqrt{2x+1}dx$$

```
>$showev('integrate(x^2*sqrt(2*x+1),x))
```

$$\int x^2\,\sqrt{2\,x+1}\,dx=\frac{(2\,x+1)^{\frac{7}{2}}}{28}-\frac{(2\,x+1)^{\frac{5}{2}}}{10}+\frac{(2\,x+1)^{\frac{3}{2}}}{12}$$

```
>$showev('integrate(x^2*sqrt(2*x+1),x,0,2))
```

$$\int_{0}^{2}x^2\,\sqrt{2\,x+1}\,dx=\frac{2\,5^{\frac{5}{2}}}{21}-\frac{2}{105}$$

```
>$showev('integrate(x^2*sqrt(2*x+1),x,0,2))$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.

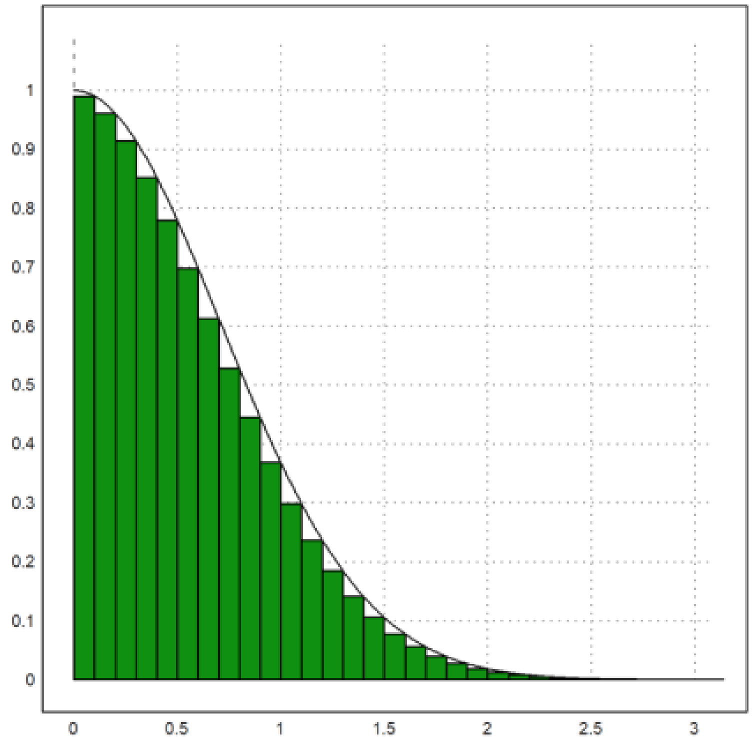
$$erf(x) = \int \frac{e^{-x^2}}{\sqrt{\pi}}\,dx.$$

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);
```

```
mendefinisikan t sebagai list untuk menyimpan nilai-nilai x
```

```
>fx &= makelist(f(t[i]+0.1),i,1,length(t));
```

simpan nilai-nilai f(x) dan jangan menggunakan x sebagai list

Hasilnya adalah:

$$\int_0^{\pi} e^{-x^2}\, dx = 0.1950839272901491$$

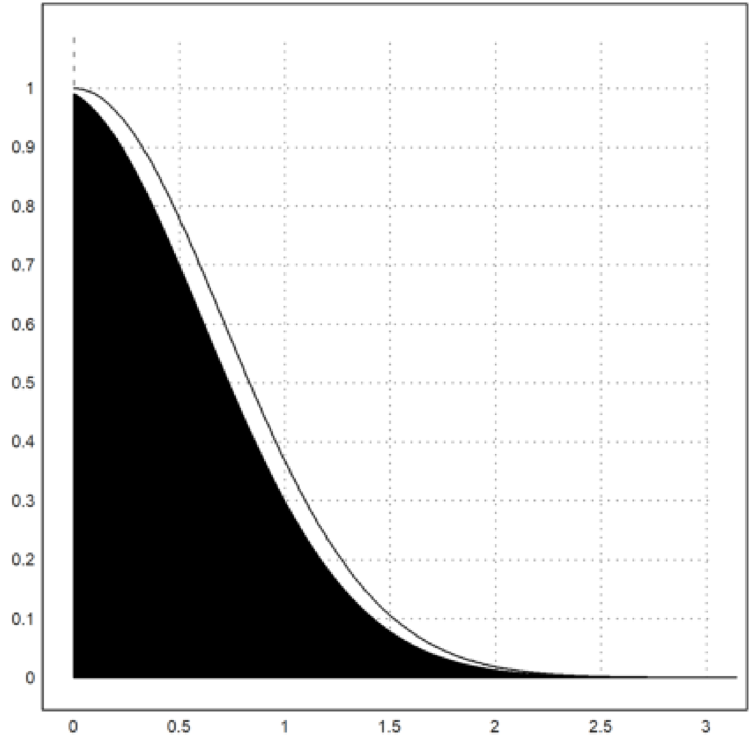
Jumlah tersebut diperoleh dari hasil kali lebar sub-subinterval (=0.1) dan jumlah nilai-nilai f(x) untuk x = 0.1, 0.2, 0.3, ..., 3.2.

```
>0.1*sum(f(x+0.1))
```

```
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.

```
>x=0:0.001:pi-0.1; plot2d(x,f(x+0.1),>bar); plot2d("f(x)",0,pi,>add):
```



Berikut adalah contoh lain fungsi yang tidak memiliki antiderivatif, sehingga integral tentunya hanya dapat dihitung dengan metode numerik.

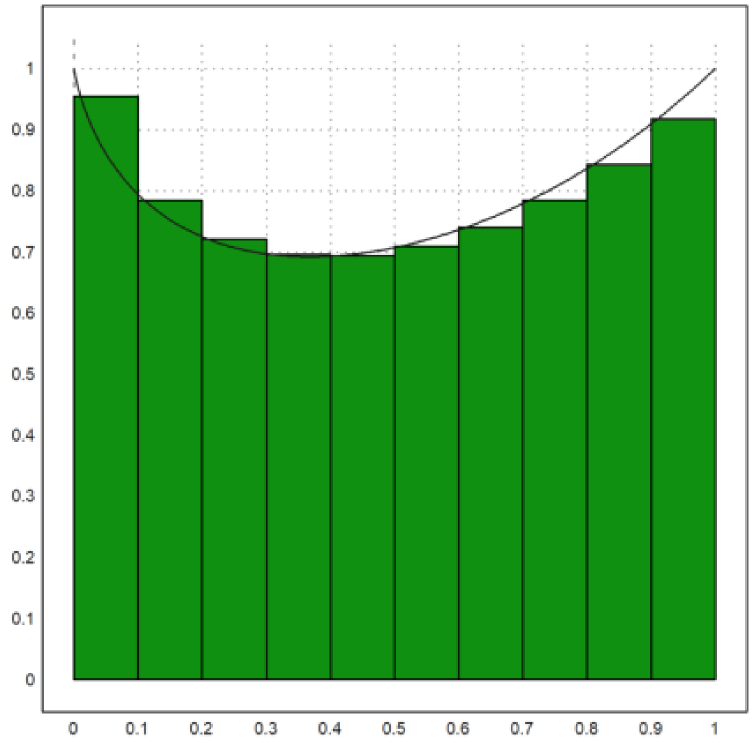
```
>function f(x) &= x^x
```

```
                                     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):
```



Karena maxima gagal menghitung integral tentu tersebut secara langsung menggunakan perintah integrate. Untuk mendapat hasil atau pendekatan nilai integral tentu tersebut kita lakukan seperti contoh sebelumnya.

```
>t &= makelist(a,a,0,1-0.01,0.01);
>fx &= makelist(f(t[i]+0.01),i,1,length(t));
```

maxima: 'integrate(f(x),x,0,1) = 0.01*sum(fx[i],i,1,length(fx))

```
>function f(x) &= sin(3*x^5+7)^2; $f(x)
```

$$\sin^2\left(3\,x^5+7\right)$$

```
>integrate(f,0,1)
```

```
0.542581176074
```

Sifat - Sifat Integral Tentu

1. Linearitas
sifat 1: Jika c adalah konstanta, maka:

$$\int_a^b cf(x)dx = c\int_a^b f(x)dx$$

Sifat 2: Jika f(x) dan g(x) adalah dua fungsi yang terintegralkan pada interval [a, b], maka:

$$\int_a^b (f(x)+g(x))dx = \int_a^b f(x)dx + \int_a^b g(x)dx$$

2. Interval

Sifat 3: Jika a < c < b, maka:

$$\int_a^b f(x)dx = \int_a^c f(x)dx + \int_c^b f(x)dx$$

3. Fungsi Genap dan Ganjil

Fungsi Genap: Jika f(x) adalah fungsi genap (yaitu, f(-x) = f(x)), maka:

$$\int_{-a}^a f(x)dx = 2\int_0^a f(x)dx$$

Fungsi Ganjil: Jika f(x) adalah fungsi ganjil (yaitu, f(-x) = -f(x)), maka:

$$\int_{-a}^a f(x)dx = 0$$

4. Batas Integral Bertukar

$$\int_b^a f(x)dx = -\int_a^b f(x)dx$$

Aplikasi Integral Tentu pada Panjang Kurva

Sebuah jalan tol memiliki bentuk lengkungan yang dapat dimodelkan dengan persamaan y = xˆ2 dari titik x = 0 hingga x = 2 (dalam kilometer). Berapakah panjang jalan tol tersebut

Penyelesaian:

$$s = \int_0^2 \sqrt{1 + (2x)^2} dx$$

```
>$showev('integrate(sqrt(1 + (2*x)^2),x,0,2))
```

$$\int_{0}^{2}{\sqrt{4\,x^2+1}\;dx}=\frac{\operatorname{asinh}4+4\,\sqrt{17}}{4}$$

```
>$float(%)
```

$$\int_{0.0}^{2.0}{\sqrt{4.0\,x^2+1.0}\;dx}=4.646783762432936$$

Hitunglah panjang kurva berikut ini dan luas daerah di dalam kurva tersebut.

$$\gamma(t) = (r(t)\cos(t), r(t)\sin(t))$$

dengan

$$r(t) = 1 + \frac{\sin(3t)}{2}, \quad 0 \le t \le 2\pi.$$

```
>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 terlebih
```

```
Syntax error in expression, or unfinished expression!
Error in:
... i,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t); >plot2d(x,y,> ...
                                                     ^
```

```
>function r(t) &= 1+sin(3*t)/2; $'r(t)=r(t)
```

$r\,([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2, 0.21, 0.2200000$

```
>function fx(t) &= r(t)*cos(t); $'fx(t)=fx(t)
```

$fx\,([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2, 0.21, 0.2200000$

```
>function fy(t) &= r(t)*sin(t); $'fy(t)=fy(t)
```

$fy\left([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2, 0.21, 0.2200000\right.$

```
>function ds(t) &= trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); $'ds(t)=ds(t)
```

```
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 numerik 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);
```

Barisan dan Deret

Barisan adalah susunan bilangan yang memiliki pola atau karakteristik tertentu, sedangkan deret adalah hasil penjumlahan dari anggota-anggota dalam barian tertentu.

Barisan dapat didefinisikan dengan beberapa cara di dalam EMT, di antaranya:
- menggunakan titik dua ":", sama seperti saat mendefinisikan vektor dengan elemen-elemen beraturan
- 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.

Contoh perhitungan barisan dan deret menggunakan EMT.

Menggunakan titik dua

```
>1:15
```

```
[1,  2,  3,  4,  5,  6,  7,  8,  9,  10,  11,  12,  13,  14,  15]
```

```
>sum(1:3:30)
```

```
145
```

```
>cumsum(1:3:30)
```

```
[1,  5,  12,  22,  35,  51,  70,  92,  117,  145]
```

Menggunakan perintah ”sequence” dan rumus barisan (suku ke -n)

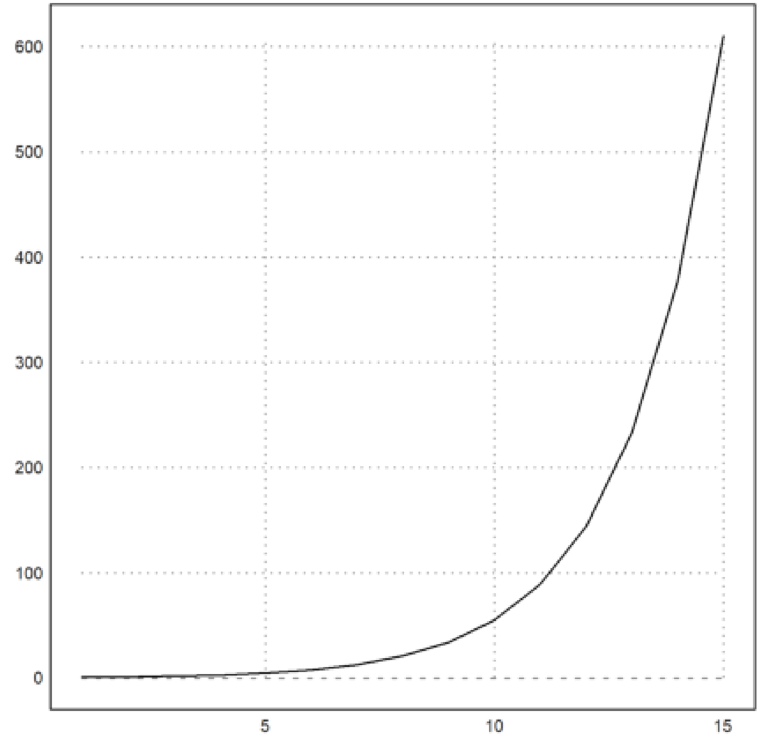
Untuk barisan yang lebih kompleks dapat digunakan fungsi ”sequence()”. Fungsi ini menghitung nilai-nilai x[n] dari semua nilai sebelumnya, x[1],...,x[n-1] yang diketahui.
Berikut adalah contoh barisan Fibonacci.

$$x_n = x_{n-1} + x_{n-2}, \quad x_1 = 1, \quad x_2 = 1$$

```
>sequence("x[n-1]+x[n-2]",[1,1],15) //suku ke satu :1, suku ke dua :1
```

```
[1,  1,  2,  3,  5,  8,  13,  21,  34,  55,  89,  144,  233,  377,  610]
```

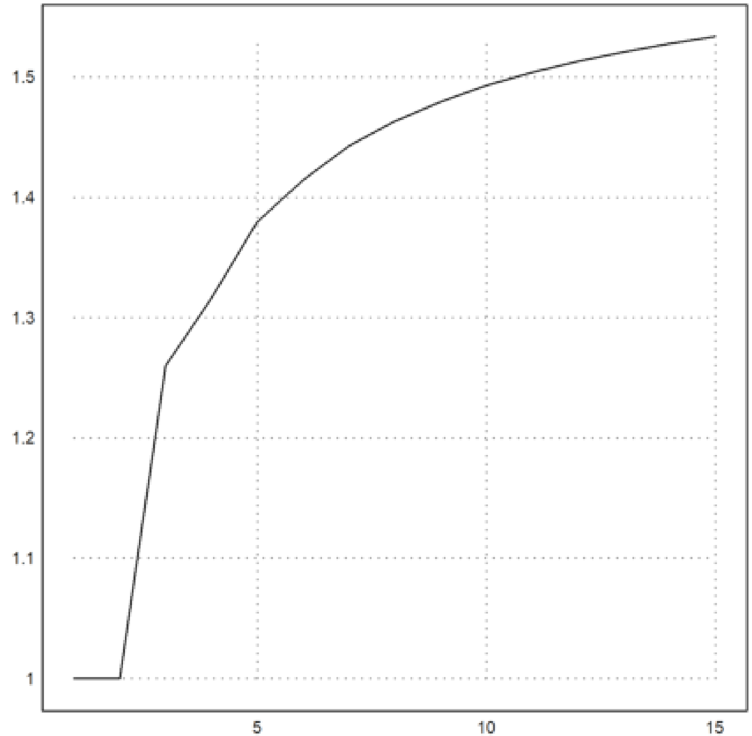
```
>plot2d(sequence("x[n-1]+x[n-2]",[1,1],15)):
```




```
>sequence("x[n-1]+x[n-2]",[1,1],15)^(1/(1:15))
```

```
[1,  1,  1.25992,  1.31607,  1.37973,  1.41421,  1.44256,  1.46311,
1.47967,  1.49292,  1.50389,  1.51309,  1.52091,  1.52765,  1.53352]
```

```
>plot2d(sequence("x[n-1]+x[n-2]",[1,1],15)^(1/(1:15))):
```



Dalam ilmu matematika, iterasi dapat diartikan sebagai proses atau metode yang digunakan secara berulang-ulang (pengulangan) dalam menyelesaikan permasalahan matematik.
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 contoh penggunaan iterasi

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
```

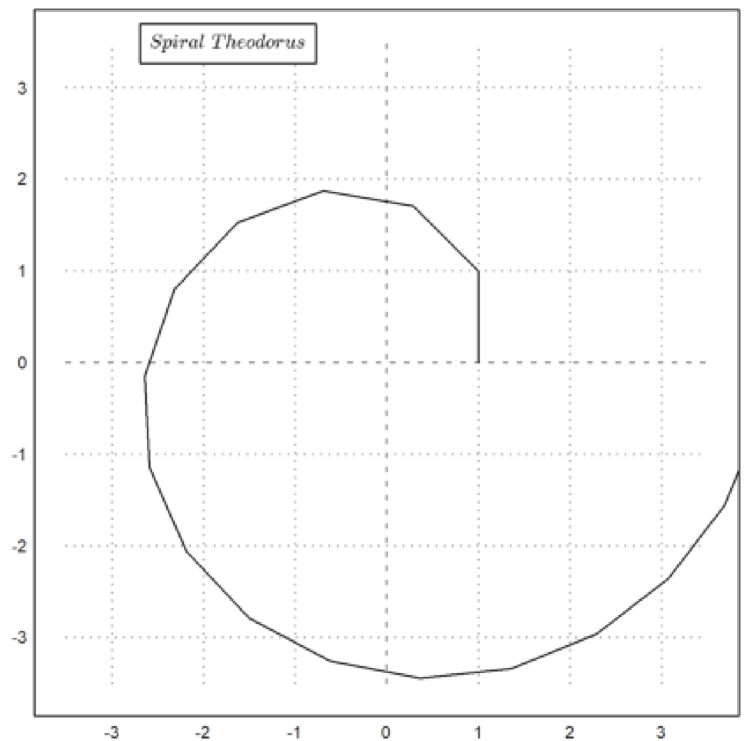
konstruksi segitiga siku-siku yang berkesinambungan menjadi spiral atau dalam kata lain sebuah spiral yang dibangun dari segitiga siku-siku yang berurutan. Setiap segitiga siku-siku yang baru dibuat dengan menghubungkan sisi miring segitiga sebelumnya dengan sisi alas yang panjangnya 1 satuan.

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)
>x=sequence("g(n-1)*x[n-1]",1,17); plot2d(x,r=3.5);...
>textbox(latex("Spiral\ Theodorus"),0.4):
```



Rekursinya dapat dijalankan sebanyak 17 untuk menghasilkan barisan 17 bilangan kompleks, kemudian digambar bilangan-bilangan kompleksnya.

Limit Barisan Limit barisan melambangkan nilai mutlak untuk bilangan

riil dan nilai modulus untuk bilangan kompleks.
Definisi formal:

$$\forall \varepsilon > 0, \exists k \in \mathbb{N} : (\forall n \in \mathbb{N}, n \geq k \implies |x_n - L| < \varepsilon)$$

Dapat dinotasikan sebagai

$$\lim_{n\to\infty} x_n = L$$

Contoh:
1.

$$\lim_{n\to\infty} \frac{1}{n}$$

```
>$showev('limit(1/n,n,inf))
```

$$\lim_{n\to\infty} \frac{1}{n} = 0$$

Kekonvergenan

Iterasi sampai konvergen merupakan proses yang terus berulang sampai mendapat nilai yang stabil atau tidak berubah secara signifikan. Tetapi, apabila iterasinya tidak konvergen setelah ditunggu lama, Kita 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.

Jika interval hasil tersebut sedikit diperlebar, akan terlihat bahwa interval tersebut memuat akar persamaan x=cos(x).

```
>h=expand(hasil,100), cos(h) << h
```

```
Variable or function hasil not found.
Error in:
h=expand(hasil,100), cos(h) << h ...
              ^
```

Deret Taylor suatu fungsi f yang diferensiabel sampai tak hingga di sekitar x=a adalah:

$$f(x) = \sum_{k=0}^{\infty} \frac{(x-a)^k f^{(k)}(a)}{k!}.$$

Diberikan contoh fungsi exponensial yang didekati dengan Deret Taylor

Menghitung exp(x) untuk x=4 yang di potong pada k=10

```
>x := 4; k := 10; hasil := round(exp(x),k)
```

```
54.5981500331
```

Bukti :

$$e^x = \frac{(x)^{10}}{(3628800)} + \frac{(x)^9}{(362880)} + \frac{(x)^8}{(40320)} + \frac{(x)^7}{(5040)} + \frac{(x)^6}{(720)} + \frac{(x)^5}{(120)} + \frac{(x)^4}{(24)} + \frac{(x)^3}{(6)} + \frac{(x)^2}{(2)} + x + 1$$

untuk x=4

$$e^4 = \frac{(4)^{10}}{(3628800)} + \frac{(4)^9}{(362880)} + \frac{(4)^8}{(40320)} + \frac{(4)^7}{(5040)} + \frac{(4)^6}{(720)} + \frac{(4)^5}{(120)} + \frac{(4)^4}{(24)} + \frac{(4)^3}{(6)} + \frac{(4)^2}{(2)} + 4 + 1$$

$$e^4 = \frac{(4)^{10}}{(3628800)} + \frac{(4)^9}{(362880)} + \frac{(4)^8}{(40320)} + \frac{(4)^7}{(5040)} + \frac{(4)^6}{(720)} + \frac{(4)^5}{(120)} + \frac{(4)^4}{(24)} + \frac{(4)^3}{(6)} + \frac{(4)^2}{(2)} + 5$$

$$e^4 = 54,4431$$

1. Tentukan suku ke-5 dari deret berikut

$$[5, 25, 125]$$

2. Hitunglah jumlah suku ke-7 dari deret yang dihasilkan no 1
3. Hitunglah limit barisan dari

$$\lim_{x \to \infty} \sqrt{\frac{4x + 1}{x}}$$

Fungsi multivariabel adalah sebuah konsep dalam matematika yang menggambarkan hubungan antara satu variabel output dengan dua atau lebih variabel input. Berbeda dengan fungsi satu variabel yang hanya melibatkan satu variabel bebas, fungsi multivariabel melibatkan beberapa variabel bebas yang secara bersama-sama menentukan nilai dari variabel terikat. Fungsi multivariabel biasanya digunakan untuk menggambarkan hubungan yang kompleks dalam bidang seperti fisika, ekonomi, teknik, dan berbagai ilmu lainnya.

Secara umum, jika f adalah sebuah fungsi multivariabel, maka bentuk umumnya dapat dinyatakan sebagai:

$$z = f(x_1, x_2, ..., x_n)$$

dimana x1,x2,...,xn adalah variabel bebas (input) menunjukkan jumlah variabel dan z adalah variable terikat (output).

contoh :
Luas Persegi Panjang: Luas persegi panjang (L) adalah fungsi dari panjang (p) dan lebar (l). Kita dapat menuliskannya sebagai L(p, l) = p * l. Di sini, L adalah variabel terikat, sedangkan p dan l adalah variabel bebas.

Diberikan fungsi:

$$x^2 + 2xy + y^2$$

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

```
 2              2
y  + 2 x y + x
```

diff(f,x): Bagian ini untuk menghitung turunan numerik.

f: Merupakan representasi dari fungsi yang ingin kita turunkan. Fungsi ini bisa berupa fungsi anonim, fungsi yang telah didefinisikan sebelumnya, atau bahkan persamaan matematika yang kompleks.

x: Menunjukkan variabel terhadap mana kita akan menurunan fungsi f. Artinya, kita akan mencari laju perubahan fungsi f terhadap perubahan nilai x.

&=: mendefinikan fungsi, seperti yang sudah dijelaskan kemarin

```
>fx &= diff(f(x,y),x)
```

```
2 y + 2 x
```

ini merupakan fungsi simbolik jadi harus didefinisikan dengan &=

Diberikan fungsi:

$$\sqrt{x} + y^2 + 2xz$$

```
>function f(x,y,z)&= sqrt(x)+y^2+2*x*z
```

```
                    2
          2 x z + y  + sqrt(x)
```

```
>fx &= diff(f(x,y,z),x)
```

```
                       1
              2 z + ---------
                    2 sqrt(x)
```

```
>fy &= diff(f(x,y,z),y)
```

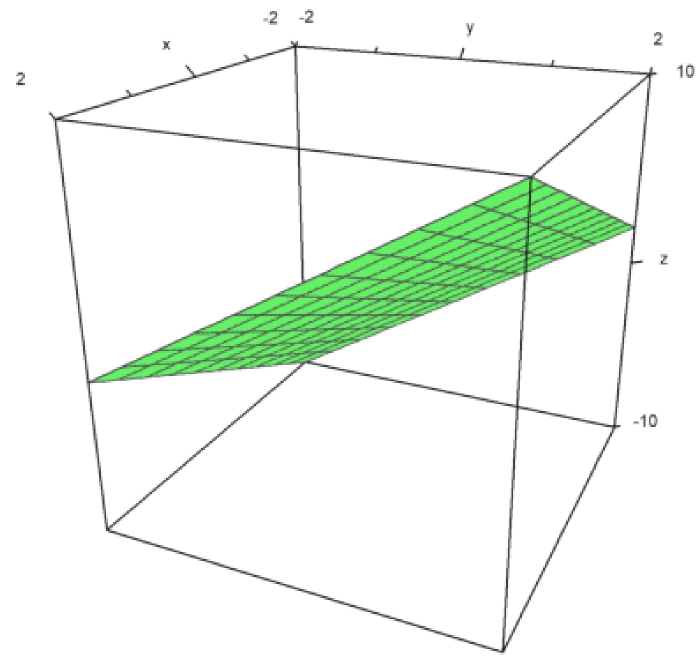
$$2\,y$$

```
>fz &= diff(f(x,y,z),z)
```

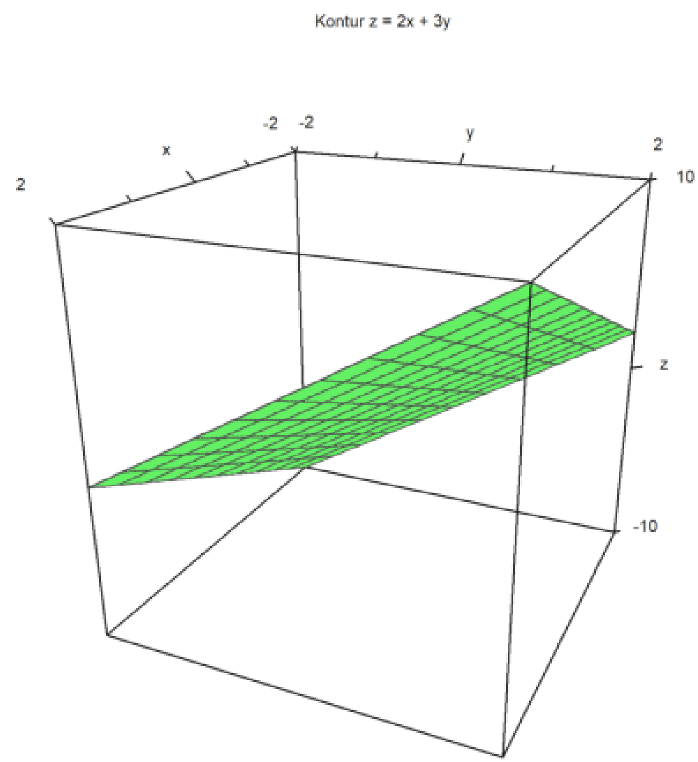
$$2\,x$$

Grafik fungsi multivariabel adalah representasi visual dari fungsi yang memiliki dua atau lebih variabel input. Untuk fungsi dua variabel $f(x,y)$, grafiknya berupa permukaan dalam ruang tiga dimensi.
Bidang z = 2x + 3y

```
>plot3d("2*x + 3*y", -2,2, -2,2):
```

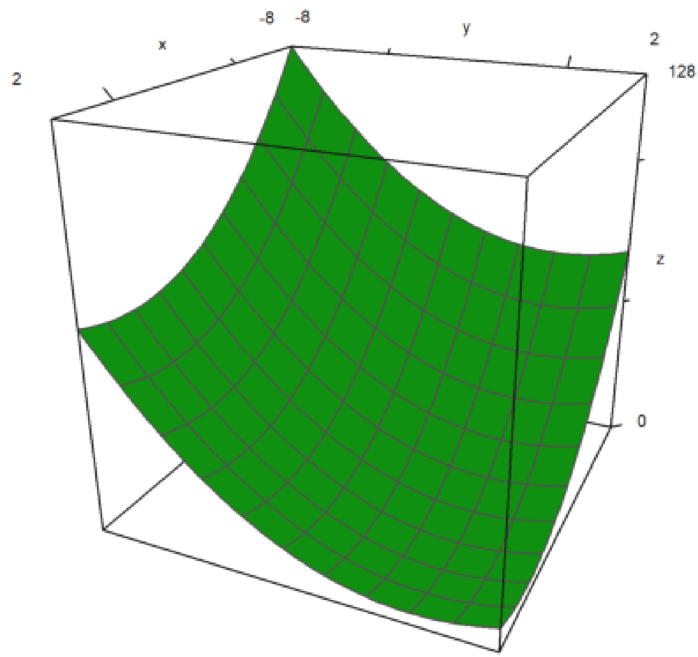



```
>title("Kontur z = 2x + 3y"):
```



Paraboloid z = xˆ2 + yˆ2

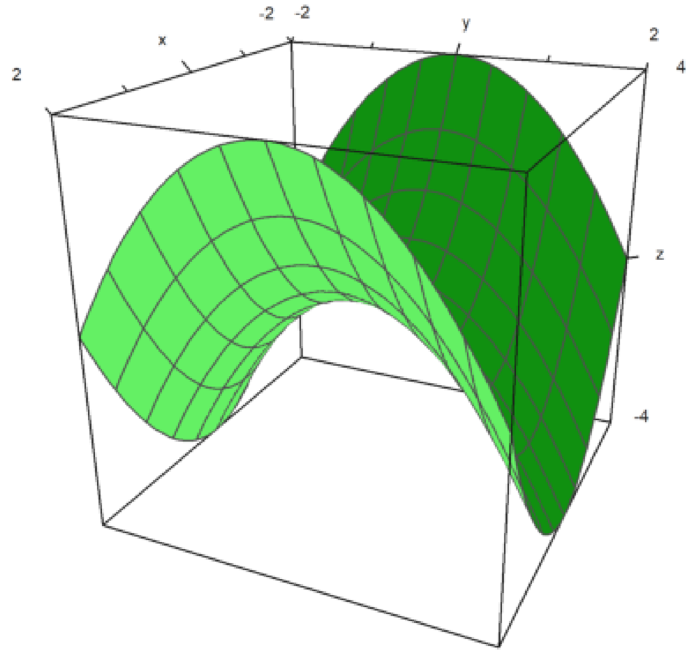
```
>plot3d("x^2 + y^2", -8,2, -8,2):
```



Hyperbolic Paraboloid

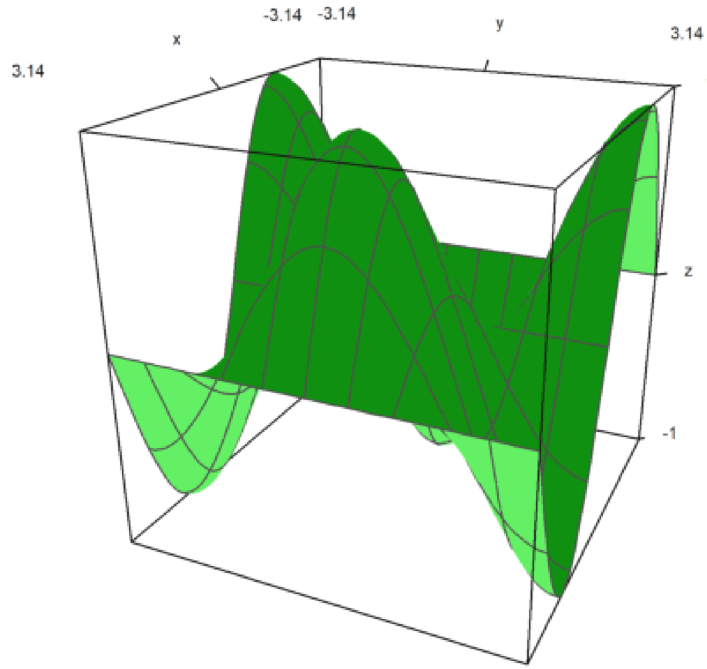
$$z = x^2 - y^2$$

```
>plot3d("x^2 - y^2", -2,2, -2,2):
```



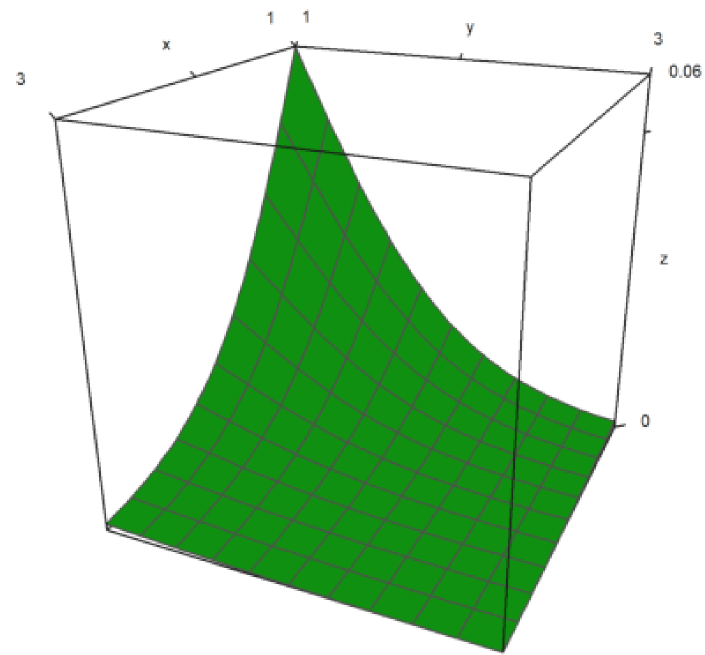
z = sin(x)cos(y)

```
>plot3d("sin(x)*cos(y)", -pi,pi, -pi,pi):
```



Distribusi normal bivariat

```
>plot3d("(1/(2*pi))*exp(-(x^2+y^2)/2)", 1,3, 1,3):
```




```
>title("Distribusi Normal Bivariat")
```

Turunan fungsi multivariabel merupakan perluasan konsep turunan dari fungsi satu variabel ke fungsi dengan dua atau lebih variabel.
menghitung turunan parsial

```
>f &= x^4 + 6*x*y + y^3
```

```
 3             4
y  + 6 x y + x
```

Integral fungsi multivariabel adalah perluasan konsep integral dari fungsi satu variabel ke fungsi dengan dua atau lebih variabel.
Contoh sederhana integral lipat dua

```
>$showev ('integrate(integrate(x^2 + y^2, x, 0, 1), y, 0, 2))
```

$$\frac{\int_{0}^{2}{3\,y^2+1\;dy}}{3}=\frac{10}{3}$$

Hasil dari integral ganda ini akan memberikan nilai total area di bawah permukaan fungsi f(x,y)=x^2+y^2 di daerah persegi yang ditentukan.

1. Misalkan medan listrik di suatu daerah diberikan oleh

$$E(x, y) = x^2 - y$$

Hitung fluks medan listrik melalui permukaan persegi panjang [0,1]×[0,2]!

```
>$showev('integrate(integrate(x^2 - y, x, 0, 1), y, 0, 2))
```

$$-\frac{\int_0^2 3\,y - 1\,dy}{3} = -\frac{4}{3}$$

kita diberikan medan listrik dalam bentuk vektor

$$E(x, y) = x^2 - y$$

dan diminta untuk menghitung fluks dari medan listrik melalui permukaan tersebut.

$$x = 0 - 1$$

$$y = 0 - 2$$

2. Misalkan sebuah perusahaan memproduksi dua jenis barang,x dan y.
Fungsi keuntungan perusahaan dinyatakan sebagai:

$$p(x, y) = 50x + 80y - 2x^2 - 3y^2 - 4xy$$

tentukan banyak barang x dan y yang harus diproduksi agar keuntungan maksimal!

```
>function p(x,y) &= 50*x+80*y-2*x^2-3*y^2-4*x*y
```

```
            2                      2
     - 3 y  - 4 x y + 80 y - 2 x  + 50 x
```

```
>grad &= [diff(p(x,y),x), diff(p(x,y),y)]=0
```

```
     [- 4 y - 4 x + 50, - 6 y - 4 x + 80] = 0
```

```
>n &= [diff(px,x), diff(py,y), diff(px,y)]
```

[0, 0, 0]

```
>&sol := 'solve(grad, n)
```

solve([- 4 y - 4 x + 50, - 6 y - 4 x + 80] = 0, [0, 0, 0])

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 poin dan plot.

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

Kemudian untuk tiga segmen.

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

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

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

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

Hitung garis tegak lurus melalui A pada BC.

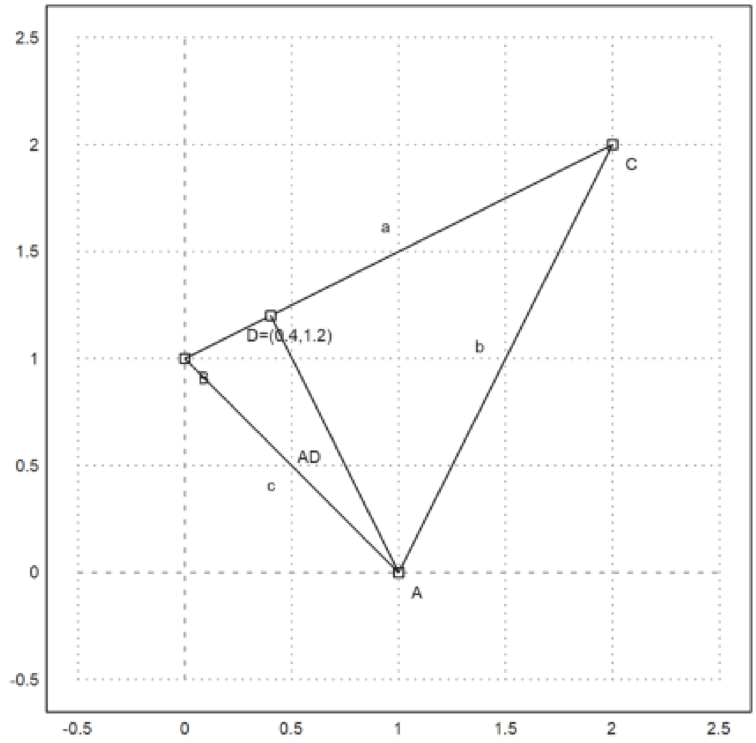
```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan persimpangan dengan BC.

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

Plot that.

```
>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.

```
>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
```

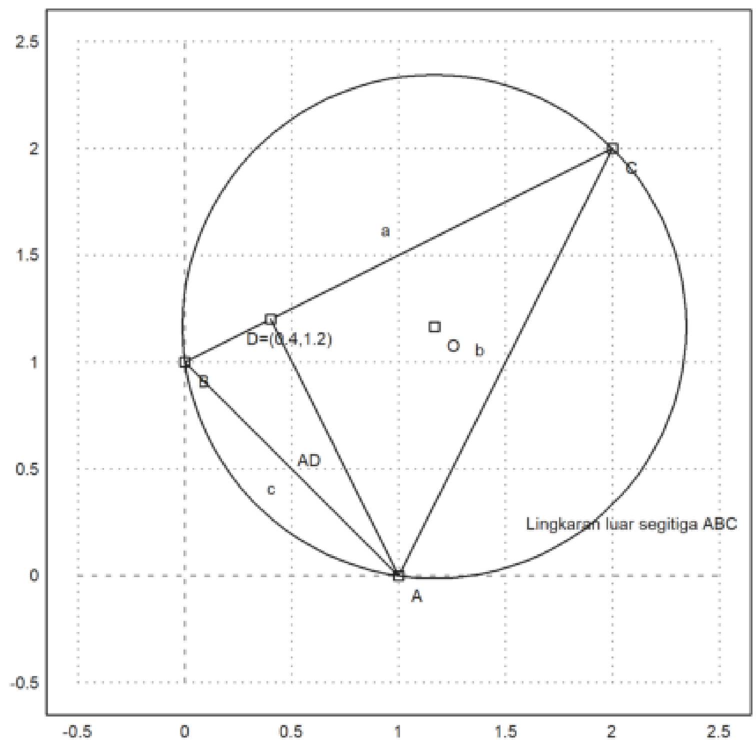
Sudut di C.

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

```
36°52'11.63''
```

Kemudian menggambarkan lingkaran luar segitiga.

```
>c=circleThrough(A,B,C); // lingkaran luar segitiga ABC
>R=getCircleRadius(c); // jari2 lingkaran luar
>O=getCircleCenter(c); // titik pusat lingkaran c
>plotPoint(O,"O"); // gambar titik "O"
>plotCircle(c,"Lingkaran luar segitiga ABC"):
```



Tampilkan koordinat titik pusat dan jari-jari lingkaran luar.

```
>O, R
```

```
[1.16667,  1.16667]
1.17851130198
```

Sekarang akan digambar lingkaran dalam segitiga ABC. Titik pusat lingkaran dalam adalah titik potong garis-garis bagi sudut.

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

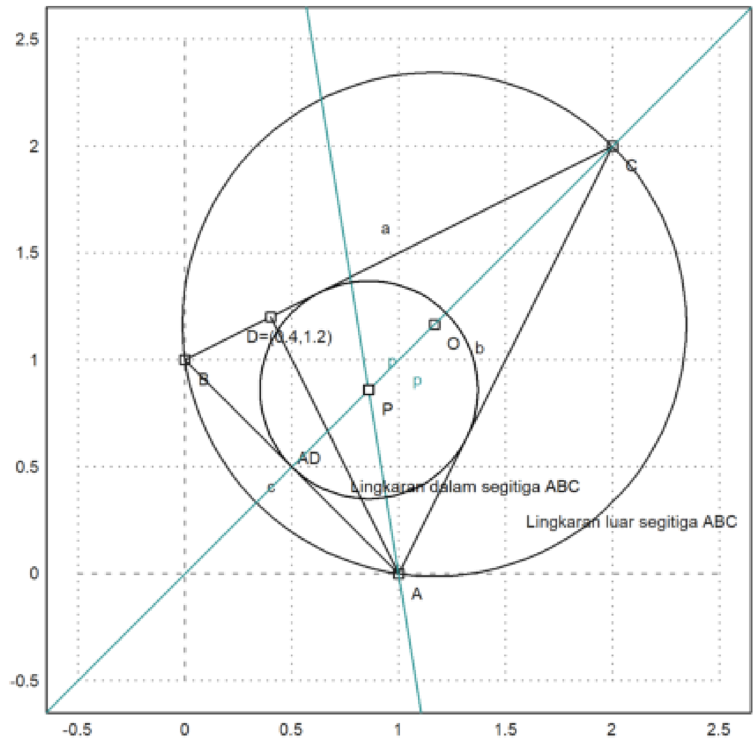
```
[0.86038,  0.86038]
```

Tambahkan semuanya ke plot.

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

```
0.509653732104
```

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

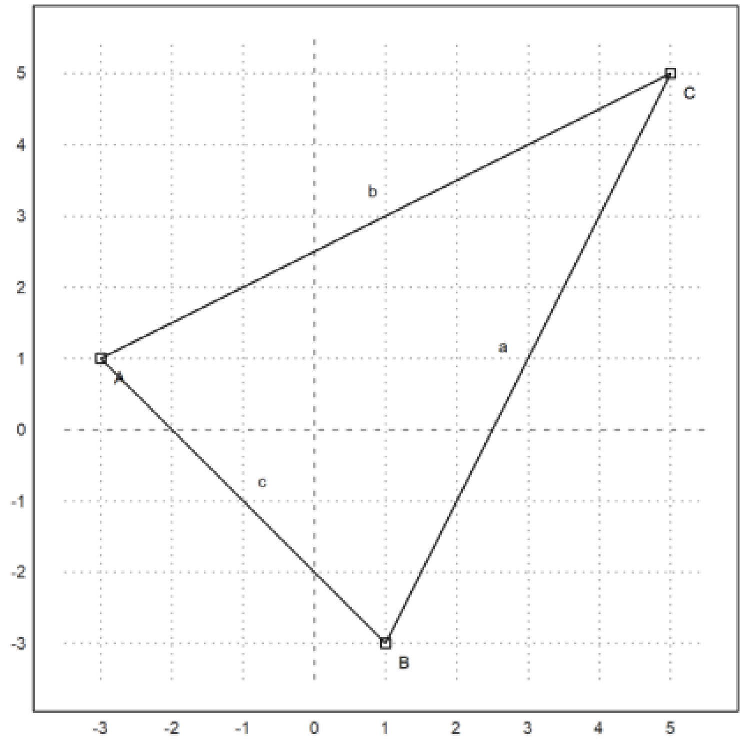


1. Tentukan ketiga titik singgung lingkaran dalam dengan sisi-sisi segitiga ABC.

```
>setPlotRange(-3.5,5.5,-3.5,5.5);
>A=[-3,1]; plotPoint(A,"A");
>B=[1,-3]; plotPoint(B,"B");
>C=[5,5]; plotPoint(C,"C");
```

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

```
>plotSegment(A,B,"c")
>plotSegment(B,C,"a")
>plotSegment(A,C,"b")
>aspect(1):
```

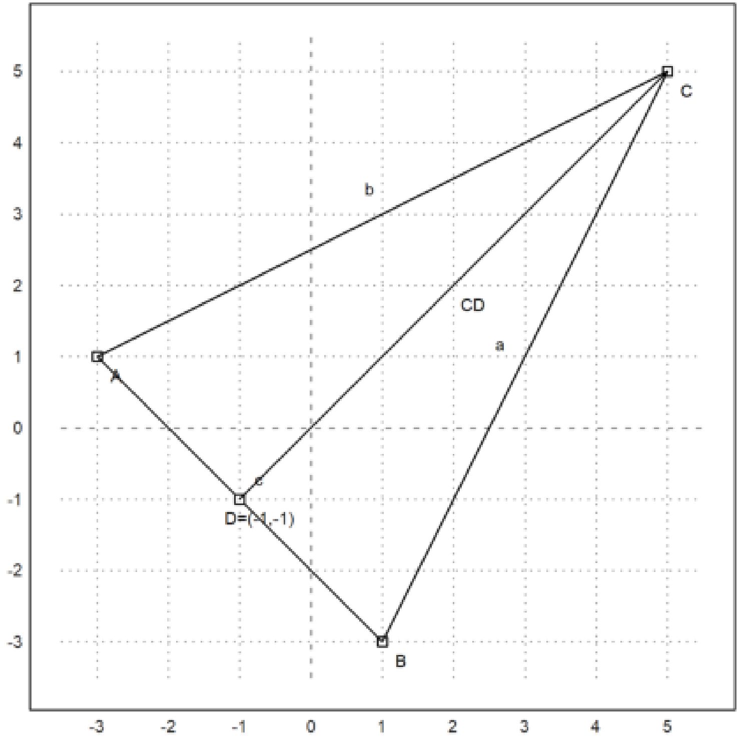


3. Hitung luas segitiga tersebut.

```
>lineThrough(A,B)
```

```
[4,  4,  -8]
```

```
>h=perpendicular(C,lineThrough(A,B));
>D=lineIntersection(h,lineThrough(A,B));
>plotPoint(D,value=1);
>aspect(1); plotSegment(C,D):
```



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

```
24
```

Jadi luas segitiga di atas adalah 24

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

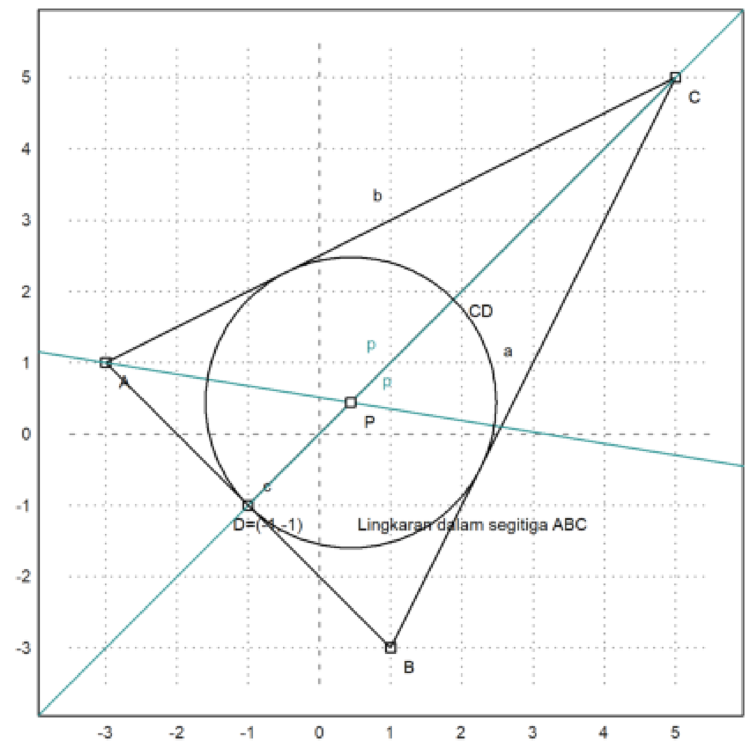
```
>l=angleBisector(A,C,B);
>g=angleBisector(C,A,B);
>P=lineIntersection(l,g)
```

```
[0.441518,  0.441518]
```

```
>color(5); plotLine(l); plotLine(g); color(1);
>plotPoint(P,"P");
>r=norm(P-projectToLine(P,lineThrough(A,B)))
```

```
2.03861492842
```

```
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"):
```



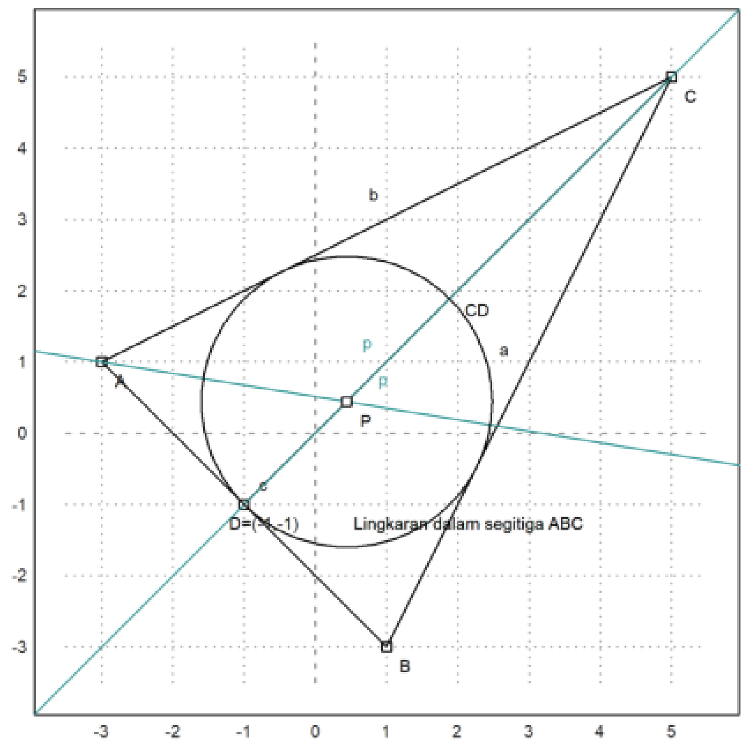
Jadi, terbukti bahwa garis bagi sudut yang ketiga juga melalui titik pusat lingkaran dalam.

5. Gambar jari-jari lingkaran dalam.

```
>r=norm(P-projectToLine(P,lineThrough(A,B)))
```

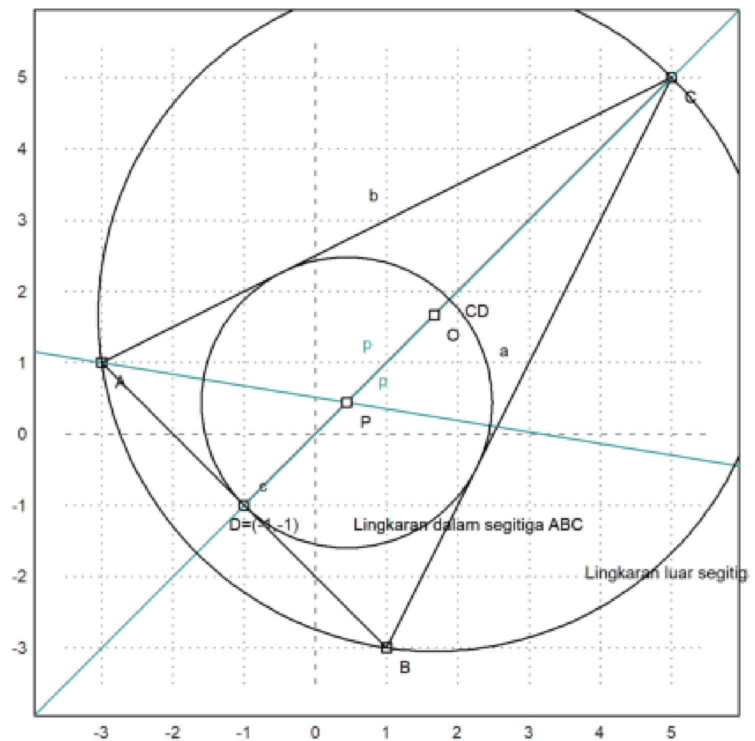
2.03861492842

```
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"):
```



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

```
>c=circleThrough(A,B,C);
>R=getCircleRadius(c);
>O=getCircleCenter(c);
>plotPoint(O,"O");
>plotCircle(c,"Lingkaran luar segitiga ABC"):
```



Contoh 2: Geometri Simbolik

Kita dapat menghitung geometri eksak dan simbolik menggunakan Maxima.

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

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

Fungsi untuk garis dan lingkaran bekerja seperti fungsi Euler, tetapi memberikan perhitungan simbolis.

```
>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
```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$$1.178511301977579$$

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

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

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

$$y = x$$

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

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

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

```
[0.86038,  0.86038]
```

Garis dan Lingkaran yang Berpotongan

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

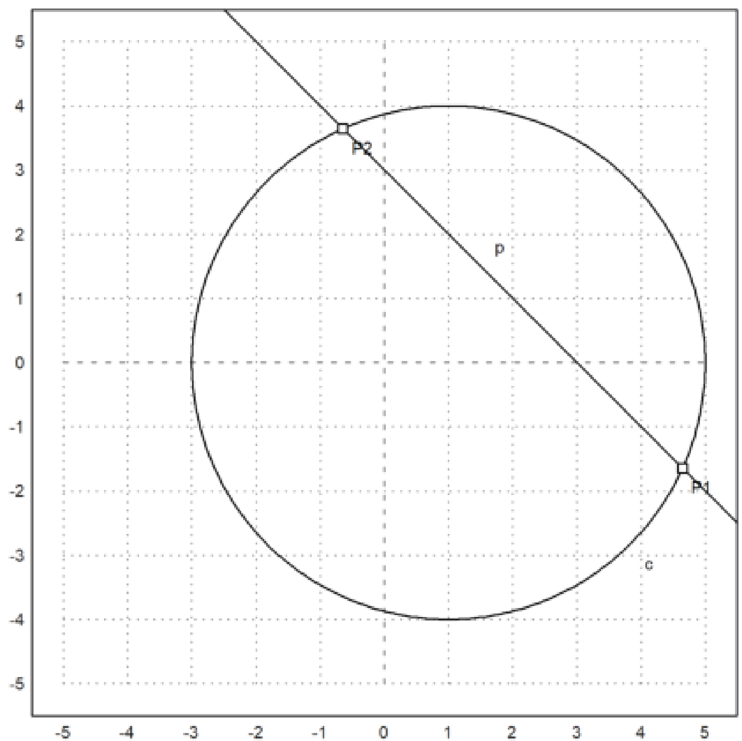
```
>A &:= [1,0]; c=circleWithCenter(A,4);
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);
>setPlotRange(5); plotCircle(c); plotLine(l);
```

Persimpangan garis dengan lingkaran mengembalikan dua titik dan jumlah titik persimpangan.

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

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

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



Begitu pula di Maxima.

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

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

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

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

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

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

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

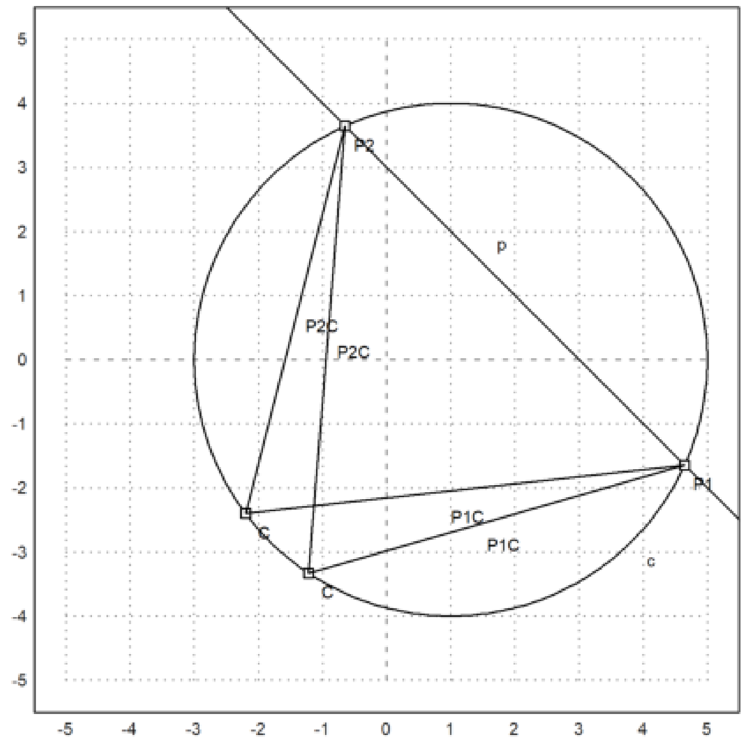
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);
>degprint(computeAngle(P1,C,P2))
```

```
69°17'42.68''
```

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

```
69°17’42.68’’
```

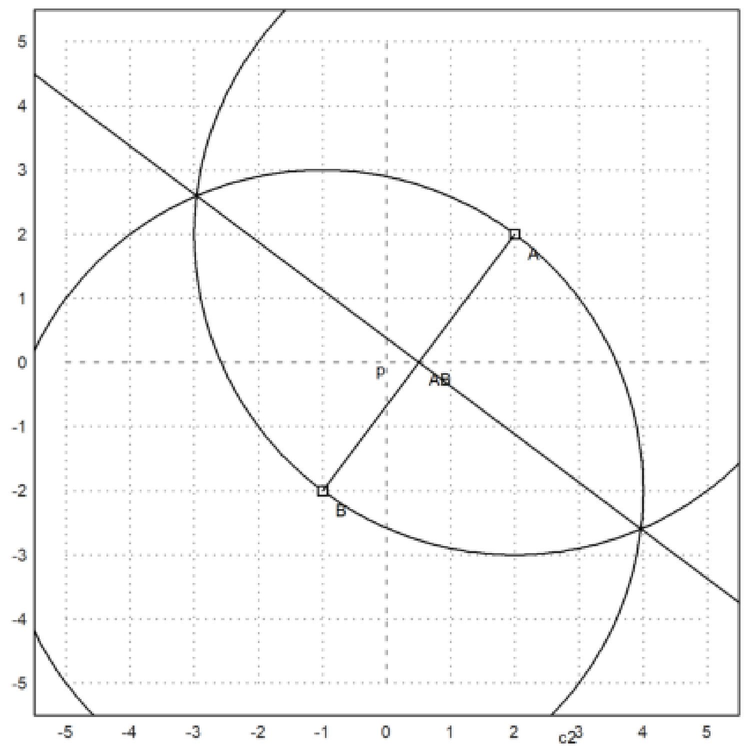
```
>insimg;
```



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, kami 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 terlibat. Tetapi kita dapat menyederhanakannya, jika kita memecahkan y.

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

$$\left[y=\frac{-\left(2\,b_{1}-2\,a_{1}\right)\,x+{b_{2}}^2+{b_{1}}^2-{a_{2}}^2-{a_{1}}^2}{2\,b_{2}-2\,a_{2}}\right]$$

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

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

$$\left[y=\frac{-\left(2\,b_{1}-2\,a_{1}\right)\,x+{b_{2}}^2+{b_{1}}^2-{a_{2}}^2-{a_{1}}^2}{2\,b_{2}-2\,a_{2}}\right]$$

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

$$\left[y=\frac{\left(b_{2}-a_{2}\right)\,x-a_{1}\,b_{2}+a_{2}\,b_{1}}{b_{1}-a_{1}}\right]$$

Perhatikan hasil kali gradien garis g dan h adalah:

$$\frac{-(b_1 - a_1)}{(b_2 - a_2)} \times \frac{(b_2 - a_2)}{(b_1 - a_1)} = -1.$$

Artinya kedua garis tegak lurus.

Contoh 3: Rumus Heron

Rumus Heron menyatakan bahwa luas segitiga dengan panjang sisi-sisi a, b dan c adalah:

$$L = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{dengan } s = (a+b+c)/2,$$

atau bisa ditulis dalam bentuk lain:

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

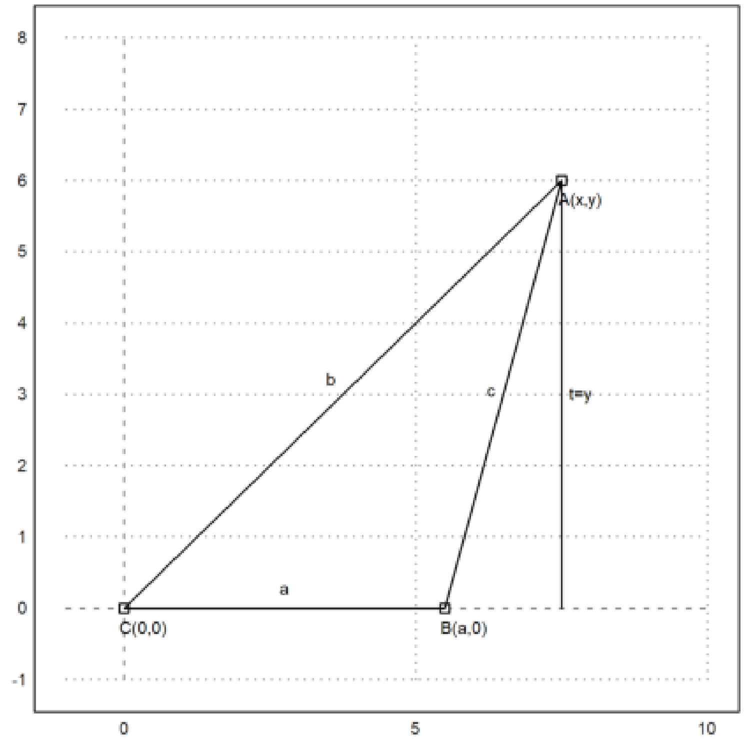
Untuk membuktikan hal ini kita misalkan C(0,0), B(a,0) dan A(x,y), b=AC, c=AB. Luas segitiga ABC adalah

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

Nilai y didapat dengan menyelesaikan sistem persamaan:

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

```
>setPlotRange(-1,10,-1,8); plotPoint([0,0], "C(0,0)"); plotPoint([5.5,0], "B(a,0)");  ...
> plotPoint([7.5,6], "A(x,y)");
>plotSegment([0,0],[5.5,0], "a",25); plotSegment([5.5,0],[7.5,6],"c",15);  ...
>plotSegment([0,0],[7.5,6],"b",25);
>plotSegment([7.5,6],[7.5,0],"t=y",25):
```




```
>sol &= solve([x^2+y^2=b^2,(x-a)^2+y^2=c^2],[x,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 formula 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 persegi panjang adalah kasus yang terkenal.

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

$$\frac{3\ |ysol|}{2}$$

Dan juga jelas, bahwa ini adalah segitiga dengan luas maksimal dan dua sisi 3 dan 4.

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

```
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 temukan himpunan semua titik di mana b+c=d untuk beberapa 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 ...
                         ^
```

And make functions of this.

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

$$0$$

$$0$$

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

Kita dapat memeriksa persamaan umum untuk elips ini, i.e.

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

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

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

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

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

Kita mendapatkan beberapa minimum, yang termasuk segitiga dengan satu sisi 0, dan solusi a=b=c=d/3.

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

Ada juga metode Lagrange, memaksimalkan H(a,b,c)^2 sehubungan dengan 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

First set the points in 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
```

Kemudian atur rentang plot, dan plot poinnya.

```
>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 the segments.

```
>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 ...
                        ^
```

CompHitung tengah tegak lurus di Maxima.

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

Dan pusat keliling.

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

Kita mendapatkan rumus untuk jari-jari lingkaran melingkar.

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

Mari kita tambahkan ini ke 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 mendapatkan rumus sederhana

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

untuk radius. Kita dapat memeriksa, apakah ini benar-benar benar dengan Maxima. Maxima akan memperhitungkan ini hanya jika kita mengkutudududukkannya.

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

Contoh 4: Garis Euler dan Parabola

Garis Euler adalah garis yang ditentukan dari segitiga apa pun yang tidak sama sisi. Ini adalah garis tengah segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga, termasuk ortosentrum, pusat lilitan, centroid, titik Exeter dan pusat lingkaran sembilan titik segitiga.

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

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

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

Untuk memplot objek geometris, kita mengatur area plot, dan menambahkan titik-titik ke dalamnya. Semua plot objek geometris ditambahkan ke plot saat ini.

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

Kita juga bisa menambahkan sisi segitiga.

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

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

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

Kita dapat menghitung koefisien sisi c.

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

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

Dan juga dapatkan rumus untuk baris ini.

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

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

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

Sekarang kita menghitung lingkaran lingkaran ABC.

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

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

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

Kita dapat menghitung persimpangan ketinggian di ABC (ortosent) secara numerik dengan perintah berikut.

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

Sekarang kita dapat menghitung garis Euler dari segitiga.

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

Tambahkan ke plot kita.

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

Pusat gravitasi harus berada di garis ini.

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

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

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

Fungsi termasuk fungsi untuk sudut juga.

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

```
60°15'18.43''
```

Persamaan untuk pusat incircle tidak terlalu bagus.

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

Mari kita hitung juga ekspresi untuk jari-jari lingkaran yang tertulis.

```
>r &= distance(Q,projectToLine(Q,lineThrough(A,B)))|ratsimp; $r
>LD &=  circleWithCenter(Q,r); // Lingkaran dalam
```

Mari kita tambahkan ini ke plot.

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

Parabola

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

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

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

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

Ini seharusnya menjadi beberapa fungsi, tetapi pemecah default Maxima hanya dapat menemukan solusinya, jika kita kuadratkan persamaannya. Akibatnya, kami mendapatkan solusi palsu.

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

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

Solusi pertama adalah

maxima: akar[1]

Menambahkan solusi pertama ke plot menunjukkan, bahwa itu memang jalan yang kita cari. Teorinya memberi tahu kita bahwa itu adalah parabola yang diputar.

```
>plot2d(&rhs(akar[1]),add=1):
>function g(x) &= rhs(akar[1]); $'g(x)= g(x)// fungsi yang mendefinisikan kurva di atas
>T &=[-1, g(-1)]; // ambil sebarang titik pada kurva tersebut
>dTC &= distance(T,C); $fullratsimp(dTC), $float(%) // jarak T ke C
>U &= projectToLine(T,lineThrough(A,B)); $U // proyeksi T pada garis AB
>dU2AB &= distance(T,U); $fullratsimp(dU2AB), $float(%) // jatak T ke AB
```

Ternyata jarak T ke C sama dengan jarak T ke AB. Coba Anda pilih titik T yang lain dan ulangi perhitungan-perhitungan di atas untuk menunjukkan bahwa hasilnya juga sama.

Contoh 5: Trigonometri Rasional

Ini terinspirasi oleh sebuah pembicaraan N.J.Wildberger. Dalam bukunya ”Divine Proportions”, Wildberger mengusulkan untuk mengganti gagasan klasik tentang jarak dan sudut dengan segi empat dan penyebaran. Dengan menggunakan ini, memang mungkin untuk menghindari fungsi trigonometri dalam banyak contoh, dan tetap ”rasional”.

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

Intinya adalah bahwa perhitungan rasional simbolik sering kali menghasilkan hasil yang sederhana. Sebaliknya, trigonometri klasik menghasilkan hasil trigonometri yang rumit, yang mengevaluasi perkiraan numerik saja.

```
>load geometry;
```

Untuk perkenalan pertama, kami menggunakan segitiga persegi panjang dengan proporsi Mesir yang terkenal 3, 4 dan 5. Perintah berikut adalah perintah Euler untuk memplot geometri bidang yang terkandung 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,

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

di mana wa adalah sudut pada A. Cara biasa untuk menghitung sudut ini, adalah dengan mengambil kebalikan dari fungsi sinus. Hasilnya adalah sudut yang tidak dapat dicerna, yang hanya dapat dicetak kira-kira.

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

```
36°52'11.63''
```

Trigonometri rasional mencoba menghindari hal ini.

Gagasan pertama dari trigonometri rasional adalah segi empat, yang menggantikan jarak. Faktanya, itu hanya jarak kuadrat. Berikut ini, a, b, dan c menunjukkan segi empat sisi.

Teorema Pythogoras hanya menjadi a+b=c kemudian.

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

$$25 = 25$$

Gagasan kedua dari trigonometri rasional adalah penyebaran. Spread mengukur pembukaan antar garis. Ini adalah 0, jika garisnya sejajar, dan 1, jika garisnya persegi panjang. Ini adalah kuadrat sinus dari sudut antara dua garis.

Penyebaran garis AB dan AC pada gambar di atas didefinisikan sebagaithe lines AB and AC in the image above is defined as

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

di mana a dan c adalah kuadrat dari segitiga persegi panjang dengan satu sudut di A.

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

Ini lebih mudah dihitung daripada sudut, tentu saja. Tetapi Anda kehilangan properti bahwa sudut dapat ditambahkan dengan mudah.

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

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

```
9/25
```

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

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

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

Hukum ini diimplementasikan dalam file geometri.e yang kita muat ke Euler.

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

Dalam kasus kami, kami mendapatkan

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

Mari kita gunakan crosslaw ini untuk menemukan penyebaran di A. Untuk melakukan ini, kami menghasilkan crosslaw untuk segi empat a, b, dan c, dan menyelesaikannya untuk spread sa yang tidak diketahui.

Anda dapat melakukan ini dengan tangan dengan mudah, tetapi saya menggunakan Maxima. Tentu saja, kami mendapatkan hasilnya, kami sudah memilikinya.

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

Kami sudah tahu ini. Definisi spread adalah kasus khusus dari crosslaw.

Kita juga dapat menyelesaikan ini untuk umum a, b, c. Hasilnya adalah rumus yang menghitung penyebaran sudut segitiga yang diberikan segi empat dari tiga sisi.

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

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

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

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

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

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

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

Ini adalah sudut dalam derajat.

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

```
67°47'32.44''
```

Contoh lain

Sekarang, mari kita coba contoh yang lebih lanjut.

Kami menetapkan tiga sudut segitiga sebagai berikut.

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

Dengan menggunakan Pythogoras, mudah untuk menghitung jarak antara dua titik. Pertama-tama saya menggunakan jarak fungsi file Euler untuk geometri. Fungsi jarak menggunakan geometri klasik.

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

Euler juga berisi fungsi untuk kuadran antara dua titik.

Dalam contoh berikut, karena c+b bukan a, segitiga tidak persegi panjang.

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

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

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

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

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

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

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

```
32.4711922908
```

Dengan menggunakan pensil dan kertas, kita dapat melakukan hal yang sama dengan hukum silang. Kami memasukkan kuadrat a, b, dan c ke dalam hukum silang dan menyelesaikan x.

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

Artinya, apa yang dilakukan penyebaran fungsi yang didefinisikan dalam "geometri.e".

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

Maxima mendapatkan hasil yang sama dengan menggunakan trigonometri biasa, jika kita memaksanya. Itu menyelesaikan suku sin(arccos(...)) menjadi hasil pecahan. Sebagian besar siswa tidak dapat melakukan ini.

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

Setelah kita memiliki penyebaran di B, kita dapat menghitung tinggi ha di sisi a. Ingatlah bahwa

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

by definition.

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

Gambar berikut telah diproduksi dengan program geometri C.a.R., yang dapat menggambar segi empat dan penyebaran.

gambar: (20) Rational_Geometry_CaR.png

Menurut definisi panjang ha adalah akar kuadrat dari segi empatnya.

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

Sekarang kita dapat menghitung luas segitiga. Jangan lupa, bahwa kita berurusan dengan segi empat!

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

Rumus penentu yang biasa menghasilkan hasil yang sama.

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

Rumus Heron

Sekarang, mari kita selesaikan masalah ini secara umum!

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

Pertama-tama kita menghitung spread di B untuk segitiga dengan sisi a, b, dan c. Kemudian kita menghitung luas kuadrat (”quadrea”?), faktorkan dengan Maxima, dan kita mendapatkan rumus Heron yang terkenal.

Harus diakui, ini sulit dilakukan dengan pensil dan kertas.

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

Aturan Triple Spread

Kerugian spread adalah tidak lagi sekadar menambahkan sudut yang sama.

Namun, tiga spread segitiga memenuhi aturan ”triple spread” berikut.

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

Aturan ini berlaku untuk tiga sudut mana pun yang jumlahnya mencapai 180°.

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

Sejak penyebaran

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

sama, aturan penyebaran tiga kali lipat juga benar, jika

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

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

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

Misalnya, kita dapat menghitung sebaran sudut 60°. Yaitu 3/4. Persamaan tersebut memiliki solusi kedua, di mana semua sebarannya adalah 0.

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

Sebaran 90° jelas adalah 1. Jika dua sudut dijumlahkan menjadi 90°, sebarannya memecahkan persamaan sebaran rangkap tiga dengan a,b,1. Dengan perhitungan berikut kita memperoleh a+b=1.

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

Karena sebaran 180°-t sama dengan sebaran t, rumus sebaran rangkap tiga juga berlaku, jika satu sudut adalah jumlah atau selisih dari dua sudut lainnya.

Jadi kita dapat menemukan sebaran sudut yang digandakan. Perhatikan bahwa ada dua solusi lagi. Kita buat ini menjadi fungsi.

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

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

Garis Bagi Sudut

Kita sudah tahu situasinya seperti ini.

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

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

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

Jadi pertama-tama kita hitung sebaran sudut yang dibagi dua di A, menggunakan rumus sebaran rangkap tiga.

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

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

Mari kita periksa persegi panjang Mesir.

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

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

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

```
18°26'5.82''
```

Titik P merupakan perpotongan garis bagi sudut dengan sumbu y.

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

```
[0,  1.33333]
```

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

Mari kita periksa sudut-sudut pada contoh spesifik kita.

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

```
0.321750554397
0.321750554397
```

Sekarang kita hitung panjang garis bagi AP.

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

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

berlaku di sembarang segitiga. Kuadratkan, itu diterjemahkan menjadi apa yang disebut "hukum sebaran"

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

di mana a, b, c menunjukkan kuadran.

Karena CPA sebarannya adalah 1-sa2, kita memperoleh bisa/1=b/(1-sa2) dan dapat menghitung bisa (kuadran garis bagi sudut).

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

Mari kita periksa rumus ini untuk nilai-nilai Mesir kita.

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

```
4.21637021356
4.21637021356
```

Kita juga dapat menghitung P menggunakan rumus spread.

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

Nilainya sama dengan yang kita dapatkan dengan rumus trigonometri.

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

```
1.33333333333
```

Sudut Tali Busur

Perhatikan situasi berikut.

```
>setPlotRange(1.2); ...
>color(1); plotCircle(circleWithCenter([0,0],1)); ...
>A:=[cos(1),sin(1)]; B:=[cos(2),sin(2)]; C:=[cos(6),sin(6)]; ...
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...
>color(3); plotSegment(A,B,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a"); ...
>color(1); O:=[0,0];  plotPoint(O,"0"); ...
>plotSegment(A,O); plotSegment(B,O); plotSegment(C,O,"r"); ...
>insimg;
```

Kita dapat menggunakan Maxima untuk memecahkan rumus penyebaran rangkap tiga untuk sudut-sudut di pusat O untuk r. Dengan demikian, kita memperoleh rumus untuk jari-jari kuadrat pericircle dalam bentuk kuadran sisi-sisinya.

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

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

Kita dapat menjadikannya fungsi Euler.

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

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

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

Radiusnya memang 1.

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

```
1
```

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

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

Faktanya, sebarannya adalah b/(4r), dan kita melihat bahwa sudut tali busur b adalah setengah sudut pusat.

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

Contoh 6: Jarak Minimal pada Bidang

Catatan awal

Fungsi yang, pada titik M di bidang, menetapkan jarak AM antara titik tetap A dan M, memiliki garis datar yang agak 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 cukup sederhana: bagian atas kerucut:

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

Tentu saja minimum 0 dicapai di A.

Sekarang kita lihat fungsi MA+MB di mana A dan B adalah dua titik (tetap). Merupakan ”fakta yang diketahui” bahwa kurva level adalah elips, titik fokusnya adalah A dan B; kecuali untuk minimum AB yang konstan pada segmen [AB]:

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

Grafiknya lebih menarik:

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

Pembatasan pada garis (AB) lebih terkenal:

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

Sekarang semuanya menjadi kurang sederhana: Tidak banyak yang tahu bahwa MA+MB+MC mencapai nilai minimumnya di satu titik bidang, tetapi menentukannya tidaklah sesederhana itu:

1) Jika salah satu sudut segitiga ABC lebih dari 120° (misalkan di A), maka nilai minimumnya tercapai di titik ini (misalkan AB+AC).

Contoh:

```
>C=[-4,1];
>function d3(x,y):=d2(x,y)+sqrt((x-C[1])^2+(y-C[2])^2)
>plot3d("d3",xmin=-5,xmax=3,ymin=-4,ymax=4);
>insimg;
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A");
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

2) Namun jika semua sudut segitiga ABC kurang dari 120°, maka nilai minimumnya berada di titik F di bagian dalam segitiga, yang merupakan satu-satunya titik yang sudut-sudut sisi ABC-nya sama (masing-masing sudutnya 120°):

```
>C=[-0.5,1];
>plot3d("d3",xmin=-2,xmax=2,ymin=-2,ymax=2):
>fcontour("d3",xmin=-2,xmax=2,ymin=-2,ymax=2,hue=1,title="The Fermat point");
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

Merupakan aktivitas yang menarik untuk mewujudkan gambar di atas dengan perangkat lunak geometri; misalnya, saya mengetahui perangkat lunak yang ditulis dalam Java yang memiliki instruksi "garis kontur"...

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

Empat titik

Langkah berikutnya adalah menambahkan titik ke-4 D dan mencoba meminimalkan MA+MB+MC+MD; katakanlah Anda adalah operator TV kabel dan ingin mencari di bidang mana Anda harus meletakkan antena Anda sehingga Anda dapat menyalurkan sinyal ke empat desa dan menggunakan panjang kabel sesedikit mungkin!

```
>D=[1,1];
>function d4(x,y):=d3(x,y)+sqrt((x-D[1])^2+(y-D[2])^2)
>plot3d("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5):
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
>P=(A_B_C_D)'; plot2d(P[1],P[2],points=1,add=1,color=12);
>insimg;
```

Masih terdapat nilai minimum dan tidak tercapai di titik A, B, C, maupun D:

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

```
[0.142858,  0.142857]
```

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

Sekarang ABCD adalah persegi, kita mengharapkan 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 di jalur program.

Pertama, kita hitung jari-jari bola.

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

Kami menggunakan file geometry.e milik Euler untuk ini.

```
>load geometry;
```

Pertama dua garis membentuk kerucut.

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

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

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

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

Lalu baris ketiga.

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

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

Kita merencanakan segalanya sejauh ini.

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

Sekarang kita ambil titik umum pada sumbu y.

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

```
[0, u]
```

Hitunglah jarak ke g1.

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

Hitunglah jarak ke g.

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

Dan temukan pusat kedua lingkaran, yang jaraknya sama.

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

Ada dua solusi.

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

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

```
[0.333333,  1]
```

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

```
[0.149071,  0.447214]
```

Gambarkan lingkaran-lingkaran tersebut ke dalam gambar.

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

Plot dengan Povray

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

Pertama-tama kita memuat fungsi povray.

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

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

Kami menyiapkan suasananya dengan tepat.

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

Berikutnya kita menulis kedua bola itu ke dalam file Povray.

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

Dan kerucutnya, transparan.

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

Kita buat bidang yang dibatasi pada kerucut.

```
>gp=g();
>pc=povcone([0,0,0],0,[0,0,a],1,"");
>vp=[gp[1],0,gp[2]]; dp=gp[3];
>writeln(povplane(vp,dp,povlook(blue,0.5),pc));
```

Sekarang kita buat dua titik pada lingkaran, 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 buat dua titik tempat bola-bola tersebut menyentuh bidang. Titik-titik ini adalah fokus elips.

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

Berikutnya kita hitung perpotongan P1P2 dengan bidang.

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

Kita menghubungkan titik-titik dengan segmen garis.

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

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

```
>pcw=povcone([0,0,0],0,[0,0,a],1.01);
>pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2],1);
>writeln(povintersection([pcw,pc1],povlook(gray)));
>pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsize/2],1);
>writeln(povintersection([pcw,pc2],povlook(gray)));
```

Mulai program Povray.

```
>povend();
```

Untuk mendapatkan Anaglyph ini, kita perlu memasukkan semuanya ke dalam fungsi scene. Fungsi ini akan digunakan dua kali nanti.

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

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

```
P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]);
writeln(povpoint(P1,povlook(yellow)));
P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]);
writeln(povpoint(P2,povlook(yellow)));
P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]];
writeln(povpoint(P3,povlook(yellow)));
P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]];
writeln(povpoint(P4,povlook(yellow)));
t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-t2)*(P2-P1);
writeln(povpoint(P5,povlook(yellow)));
writeln(povsegment(P1,P2,povlook(yellow)));
writeln(povsegment(P5,P3,povlook(yellow)));
writeln(povsegment(P5,P4,povlook(yellow)));
pcw=povcone([0,0,0],0,[0,0,a],1.01);
pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2],1);
writeln(povintersection([pcw,pc1],povlook(gray)));
pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsize/2],1);
writeln(povintersection([pcw,pc2],povlook(gray)));
endfunction
```

Anda memerlukan kacamata merah/cyan untuk menghargai efek berikut.

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

Contoh 8: Geometri Bumi

Dalam buku catatan ini, kami ingin melakukan beberapa perhitungan sferis. Fungsi-fungsi tersebut terdapat dalam berkas "spherical.e" di folder contoh. Kami perlu memuat berkas tersebut terlebih dahulu.

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

Untuk memasukkan posisi geografis, kami menggunakan vektor dengan dua koordinat dalam radian (utara dan timur, nilai negatif untuk selatan dan barat). Berikut ini adalah koordinat untuk Kampus FMIPA UNY.

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

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

Anda dapat mencetak posisi ini dengan sposprint (cetak posisi bulat).

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

```
S 7°46.467’ E 110°23.050’
```

Mari kita tambahkan dua kota lagi, Solo dan Semarang.

```
>Solo=[rad(-7,-34.333),rad(110,49.683)]; Semarang=[rad(-6,-59.05),rad(110,24.533)];
>sposprint(Solo), sposprint(Semarang),
```

```
S 7°34.333' E 110°49.683'
S 6°59.050' E 110°24.533'
```

Pertama, kita hitung vektor dari satu ke yang lain pada bola ideal. Vektor ini adalah [arah, jarak] dalam radian. Untuk menghitung jarak di bumi, kita kalikan dengan jari-jari bumi pada garis lintang 7°.

```
>br=svector(FMIPA,Solo); degprint(br[1]), br[2]*rearth(7°)->km // perkiraan jarak FMIPA-Solo
```

```
65°20'26.60''
53.8945384608
```

Ini adalah perkiraan yang bagus. Rutin berikut menggunakan perkiraan yang lebih baik lagi. Pada jarak yang 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 ...
                              ^
```

Ada fungsi untuk judul, yang memperhitungkan bentuk elips bumi. Sekali lagi, kami mencetak dengan cara yang canggih.

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

```
65.34°
```

Sudut suatu segitiga melebihi 180° pada bola.

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

```
180°0'10.77''
```

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

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

```
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 ...
                                ^
```

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

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

```
2123.64310526 km^2
```

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

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

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

Fungsi-fungsi ini mengasumsikan bentuk bola yang ideal. Sama halnya di bumi.

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

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

Mari kita lihat contoh yang lebih besar, Tugu Jogja dan Monas Jakarta (menggunakan Google Earth untuk mencari koordinatnya).

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

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

Menurut Google Earth, jaraknya adalah 429,66 km. Kami memperoleh 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 ...
                         ^
```

Judulnya sama dengan yang dihitung di Google Earth.

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

```
294°17'2.85''
```

Akan tetapi, kita tidak lagi memperoleh posisi target yang tepat, jika kita menambahkan arah dan jarak ke posisi awal. Hal ini terjadi karena kita tidak menghitung fungsi invers secara tepat, tetapi mengambil perkiraan radius bumi di sepanjang lintasan.

```
>sposprint(esadd(Tugu,esdir(Tugu,Monas),esdist(Tugu,Monas)))
```

```
S 6°10.500' E 106°48.717'
```

Namun, kesalahannya tidak besar.

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

```
 S 6°10.500' E 106°48.717'
```

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

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

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

Sekarang kita tambahkan 10 dikalikan sepersepuluh jaraknya, dengan memakai arah ke Monas, kita sampai di Tugu.

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

Hasilnya sangat jauh.

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

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

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

```
>P1=[30°,10°]; P2=[30°,50°];
```

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

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

```
      79.69°
```

Namun, jika kita mengikuti pembacaan kompas ini, kita akan berputar ke kutub utara! Jadi kita harus menyesuaikan arah kita di sepanjang jalan. Untuk tujuan kasar, kita menyesuaikannya pada 1/10 dari total jarak.

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

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

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

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

```
0.203km
```

Kita memperoleh perkiraan yang baik, jika kita menyesuaikan arah setelah setiap 1/100 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 keperluan navigasi, kita bisa mendapatkan urutan posisi GPS sepanjang lingkaran besar menuju Monas dengan fungsi navigasi.

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

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

Kita menulis suatu fungsi yang memplot bumi, dua posisi, dan posisi di antaranya.

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

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

Sekarang rencanakan semuanya.

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

Atau gunakan plot3d untuk mendapatkan tampilan anaglifnya. Ini tampak sangat bagus dengan kaca mata merah/biru kehijauan.

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

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

Petunjuk:

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

```
>load geometry
```

```
Numerical and symbolic geometry.
```

```
>setPlotRange(-3.5,3.5,-3.5,3.5);
>A=[-2,-2]; plotPoint(A,"A");
>B=[2,-2]; plotPoint(B,"B");
>C=[0,3]; plotPoint(C,"C");
>plotSegment(A,B,"c");
>plotSegment(B,C,"a");
>plotSegment(A,C,"b");
>aspect(1):
>c=circleThrough(A,B,C);
>R=getCircleRadius(c);
>O=getCircleCenter(c);
```

```
>plotPoint(O,"O");
>l=angleBisector(A,C,B);
>color(2); plotLine(l); color(1);
>plotCircle(c,"Lingkaran luar segitiga ABC"):
```

2. 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.

```
>load geometry;
>setPlotRange(5); P=[2,0]; Q=[4,0]; R=[0,-4];
>plotPoint(P,"P"); plotPoint(Q,"Q"); plotPoint(R,"R"):
>sol &= solve([a+b=-c,16*a+4*b=-c,c=-4],[a,b,c])
```

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

Sehingga dapat ditentukan nilai a = -1, b = 5 dan c = -4

```
>function y&=4*x^2+5*x-12
```

```
                             2
                          4 x  + 5 x - 12
```

```
>plot2d("4*x^2+5*x-12",-13,13,-13,13):
```

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.

```
>load geometry
```

```
Numerical and symbolic geometry.
```

```
>setPlotRange(-5,5,-5,5);
>A=[-4,-4]; plotPoint(A,"A");
>B=[4,-4]; plotPoint(B,"B");
>C=[4,4]; plotPoint(C,"C");
>D=[-4,4]; plotPoint(D,"D");
>plotSegment(A,B,"");
>plotSegment(B,C,"");
>plotSegment(C,D,"");
>plotSegment(A,D,"");
>aspect(1):
>l=angleBisector(A,B,C);
>m=angleBisector(B,C,D);
>P=lineIntersection(l,m);
>color(5); plotLine(l); plotLine(m); color(1);
>plotPoint(P,"P"):
```

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

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

Dapat dilihat bahwa sisi-sisinya merupakan garis singgung lingkaran yang sama.

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

```
>AB=norm(A-B)
```

```
8
```

```
>CD=norm(C-D)
```

```
8
```

```
>AD=norm(A-D)
```

```
8
```

```
>BC=norm(B-C)
```

```
8
```

```
>AB.CD
```

```
64
```

```
>AD.BC
```

```
64
```

Terbukti bahwa hasil kali panjang sisi-sisi yang berhadapan sama yaitu 64. Jadi dapat dipastikan bahwa segiempat tersebut merupakan segiempat garis singgung.

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).

Penyelesaian :
Diketahui kedua titik fokus P = [-2,-2] dan Q = [2,-2]

```
>P=[-2,-2]; Q=[2,-2];
>function d1(x,y):=sqrt((x-P[1])^2+(y-P[2])^2)
>Q=[2,-2]; function d2(x,y):=sqrt((x-P[1])^2+(y-P[2])^2)+sqrt((x-Q[1])^2+(y-Q[2])^2)
>fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
>plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
>plot2d("abs(x+1)+abs(x-1)",xmin=-3,xmax=3):
```

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

```
>P=[-2,-2]; Q=[2,-2];
>function d1(x,y):=sqrt((x-p[1])^2+(y-p[2])^2)
>Q=[2,-2]; function d2(x,y):=sqrt((x-P[1])^2+(y-P[2])^2)+sqrt((x+Q[1])^2+(y+Q[2])^2)
>fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
>plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
>plot2d("abs(x+1)+abs(x-1)",xmin=-3,xmax=3):
>
```

EMT untuk Statistika

Dalam buku catatan ini, kami mendemonstrasikan plot statistik utama, tes dan distribusi dalam Euler.

Mari kita mulai dengan beberapa statistik deskriptif. Ini bukanlah sebuah pengantar statistik. Jadi, Anda mungkin memerlukan beberapa latar belakang untuk memahami detailnya.

Asumsikan pengukuran berikut ini. Kita 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 memplot plot kotak dan kumis untuk data tersebut. Dalam kasus kami, tidak ada pencilan.

```
>aspect(1.75); boxplot(M):
```

Kami menghitung probabilitas bahwa suatu nilai lebih besar dari 1005, dengan mengasumsikan nilai yang diukur dari distribusi normal.

Semua fungsi untuk distribusi dalam Euler diakhiri dengan ...dis dan menghitung distribusi probabilitas kumulatif (CPF).

$$\text{normaldis(x,m,d)} = \int_{-\infty}^{x} \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t-m}{d})^2}\ dt.$$

Kami mencetak hasilnya dalam % dengan akurasi 2 digit menggunakan fungsi cetak.

```
>print((1-normaldis(1005,mean(M),dev(M)))*100,2,unit=" %")
```

```
3.07 %
```

Untuk contoh berikutnya, kami mengasumsikan jumlah pria berikut ini dalam rentang ukuran tertentu.

```
>r=155.5:4:187.5; v=[22,71,136,169,139,71,32,8];
```

Berikut ini adalah plot distribusinya.

```
>plot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style="\/"):
```

Kita dapat memasukkan data mentah tersebut ke dalam tabel.

Tabel adalah sebuah metode untuk menyimpan data statistik. Tabel kita harus berisi tiga kolom: Awal rentang, akhir rentang, jumlah orang dalam rentang.

Tabel dapat dicetak dengan header. Kami menggunakan vektor string untuk mengatur header.

```
>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 lain dari ukuran, kita perlu menghitung titik tengah rentang. Kita dapat menggunakan dua kolom pertama dari tabel kita untuk hal ini.

Sumbol “|” digunakan untuk memisahkan kolom, fungsi “writetable” digunakan untuk menulis tabel, dengan opsi “labc” untuk menentukan judul kolom.

```
>(T[,1]+T[,2])/2 // the midpoint of each interval
```

```
157.5
161.5
165.5
169.5
173.5
177.5
181.5
185.5
```

Tetapi akan lebih mudah, untuk melipat rentang 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 rata-rata dan deviasi sampel dengan frekuensi yang diberikan.

```
>{m,d}=meandev(M,v); m, d,
```

```
169.901234568
5.98912964449
```

Mari kita tambahkan distribusi normal dari nilai-nilai tersebut ke dalam diagram 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 nilainya antara 0 dan 1, untuk memplotnya pada diagram batang, nilai tersebut harus dikalikan dengan 4 kali jumlah data.

```
>plot2d("qnormal(x,m,d)*sum(v)*4", ...
>  xmin=min(r),xmax=max(r),thickness=3,add=1):
```

Dalam direktori buku catatan ini, Anda dapat menemukan file dengan tabel. Data tersebut mewakili hasil survei. Berikut adalah empat baris pertama dari file tersebut. Data berasal dari buku online berbahasa Jerman "Einführung in die Statistik mit R" oleh A. Handl.

```
>printfile("table.dat",4);
```

```
Person Sex Age Titanic Evaluation Tip Problem
1 m 30 n . 1.80 n
2 f 23 y g 1.80 n
3 f 26 y g 1.80 y
```

Tabel berisi 7 kolom angka atau token (string). Kita ingin membaca tabel tersebut dari file. Pertama, kita menggunakan terjemahan kita sendiri untuk token-token tersebut.

Untuk itu, kita mendefinisikan set token. Fungsi strtokens() mendapatkan vektor string token dari string yang diberikan.

```
>mf:=["m","f"]; yn:=["y","n"]; ev:=strtokens("g vg m b vb");
```

Sekarang kita membaca tabel dengan terjemahan ini.

Argumen tok2, tok4, dan lain-lain adalah terjemahan dari kolom-kolom tabel. Argumen-argumen ini tidak ada dalam daftar parameter readtable(), jadi Anda harus menyediakannya dengan ":=".

```
>{MT,hd}=readtable("table.dat",tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
>load over statistics;
```

Untuk mencetak, kita perlu menentukan set token yang sama. Kami mencetak empat baris pertama saja.

```
>writetable(MT[1:10],labc=hd,wc=5,tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
```

```
  Person  Sex  Age Titanic Evaluation  Tip Problem
       1    m   30       n          .  1.8       n
       2    f   23       y          g  1.8       n
       3    f   26       y          g  1.8       y
       4    m   33       n          .  2.8       n
       5    m   37       n          .  1.8       n
       6    m   28       y          g  2.8       y
       7    f   31       y         vg  2.8       n
       8    m   23       n          .  0.8       n
       9    f   24       y         vg  1.8       y
      10    m   26       n          .  1.8       n
```

Tanda titik “.” mewakili nilai yang tidak tersedia.

Jika kita tidak ingin menentukan token untuk terjemahan sebelumnya, kita hanya perlu menentukan kolom mana yang berisi token dan bukan angka.

```
>ctok=[2,4,5,7]; {MT,hd,tok}=readtable("table.dat",ctok=ctok);
```

Fungsi readtable() sekarang mengembalikan satu set token.

```
>tok
```

```
m
n
f
y
g
vg
```

Tabel berisi entri dari file dengan token yang diterjemahkan ke angka.

String khusus NA=“.” ditafsirkan sebagai “Tidak Tersedia”, dan mendapatkan NAN (bukan angka) dalam tabel. Terjemahan ini dapat diubah dengan parameter NA, dan NAval.

```
>MT[1]
```

```
[1,  1,  30,  2,  NAN,  1.8,  2]
```

Berikut ini adalah isi tabel dengan angka yang tidak diterjemahkan.

```
>writetable(MT,wc=5)
```

```
    1    1   30    2    .  1.8    2
    2    3   23    4    5  1.8    2
    3    3   26    4    5  1.8    4
    4    1   33    2    .  2.8    2
    5    1   37    2    .  1.8    2
    6    1   28    4    5  2.8    4
    7    3   31    4    6  2.8    2
    8    1   23    2    .  0.8    2
    9    3   24    4    6  1.8    4
   10    1   26    2    .  1.8    2
   11    3   23    4    6  1.8    4
   12    1   32    4    5  1.8    2
   13    1   29    4    6  1.8    4
   14    3   25    4    5  1.8    4
   15    3   31    4    5  0.8    2
   16    1   26    4    5  2.8    2
   17    1   37    2    .  3.8    2
   18    1   38    4    5    .    2
   19    3   29    2    .  3.8    2
   20    3   28    4    6  1.8    2
   21    3   28    4    1  2.8    4
   22    3   28    4    6  1.8    4
   23    3   38    4    5  2.8    2
   24    3   27    4    1  1.8    4
   25    1   27    2    .  2.8    4
```

Untuk kenyamanan, Anda dapat menaruh output dari readtable() ke dalam sebuah daftar.

```
>Table={{readtable("table.dat",ctok=ctok)}};
```

Dengan menggunakan kolom token yang sama dan token yang dibaca dari file, kita dapat mencetak tabel. Kita dapat menentukan ctok, tok, dll. atau menggunakan daftar Tabel.

```
>writetable(Table,ctok=ctok,wc=5);
```

```
  Person  Sex  Age Titanic Evaluation  Tip Problem
       1    m   30       n          .  1.8       n
       2    f   23       y          g  1.8       n
       3    f   26       y          g  1.8       y
       4    m   33       n          .  2.8       n
       5    m   37       n          .  1.8       n
       6    m   28       y          g  2.8       y
       7    f   31       y         vg  2.8       n
       8    m   23       n          .  0.8       n
       9    f   24       y         vg  1.8       y
      10    m   26       n          .  1.8       n
      11    f   23       y         vg  1.8       y
      12    m   32       y          g  1.8       n
      13    m   29       y         vg  1.8       y
      14    f   25       y          g  1.8       y
      15    f   31       y          g  0.8       n
      16    m   26       y          g  2.8       n
      17    m   37       n          .  3.8       n
      18    m   38       y          g    .       n
```

```
   19    f    29        n          .  3.8       n
   20    f    28        y         vg  1.8       n
   21    f    28        y          m  2.8       y
   22    f    28        y         vg  1.8       y
   23    f    38        y          g  2.8       n
   24    f    27        y          m  1.8       y
   25    m    27        n          .  2.8       y
```

Fungsi tablecol() mengembalikan nilai kolom dari tabel, melewatkan setiap baris dengan nilai NAN (“.” dalam file), dan indeks kolom, yang berisi nilai-nilai ini.

```
>{c,i}=tablecol(MT,[5,6]);
```

Kita dapat menggunakan ini untuk mengekstrak kolom dari tabel untuk tabel baru.

```
>j=[1,5,6]; writetable(MT[i,j],labc=hd[j],ctok=[2],tok=tok)
```

```
  Person Evaluation       Tip
       2          g       1.8
       3          g       1.8
       6          g       2.8
       7         vg       2.8
       9         vg       1.8
      11         vg       1.8
      12          g       1.8
      13         vg       1.8
      14          g       1.8
```

```
15          g       0.8
16          g       2.8
20         vg       1.8
21          m       2.8
22         vg       1.8
23          g       2.8
24          m       1.8
```

Tentu saja, kita perlu mengekstrak tabel itu sendiri dari daftar Tabel dalam kasus ini.

```
>MT=Table[1];
```

Tentu saja, kita juga dapat menggunakannya untuk menentukan nilai rata-rata kolom atau nilai statistik lainnya.

```
>mean(tablecol(MT,6))
```

```
2.175
```

Fungsi getstatistics() mengembalikan elemen-elemen dalam sebuah vektor, dan jumlahnya. Kita menerapkannya pada nilai “m” dan “f” pada kolom kedua tabel kita.

```
>{xu,count}=getstatistics(tablecol(MT,2)); xu, count,
```

```
[1,  3]
[12,  13]
```

Kita bisa mencetak hasilnya dalam tabel baru.

```
>writetable(count',labr=tok[xu])
```

```
          m         12
          f         13
```

Fungsi selecttable() mengembalikan sebuah tabel baru dengan nilai dalam satu kolom yang dipilih dari vektor indeks. Pertama, kita mencari indeks dari dua nilai kita dalam tabel token.

```
>v:=indexof(tok,["g","vg"])
```

```
[5,  6]
```

Sekarang kita dapat memilih baris-baris dari tabel, yang memiliki salah satu nilai dalam v di baris ke-5.

```
>MT1:=MT[selectrows(MT,5,v)]; i:=sortedrows(MT1,5);
```

Sekarang kita dapat mencetak tabel, dengan nilai yang diekstrak dan diurutkan di kolom ke-5.

```
>writetable(MT1[i],labc=hd,ctok=ctok,tok=tok,wc=7);
```

```
 Person    Sex    Age Titanic Evaluation    Tip Problem
      2      f     23       y          g    1.8       n
      3      f     26       y          g    1.8       y
      6      m     28       y          g    2.8       y
     18      m     38       y          g      .       n
     16      m     26       y          g    2.8       n
     15      f     31       y          g    0.8       n
     12      m     32       y          g    1.8       n
     23      f     38       y          g    2.8       n
     14      f     25       y          g    1.8       y
      9      f     24       y         vg    1.8       y
      7      f     31       y         vg    2.8       n
     20      f     28       y         vg    1.8       n
     22      f     28       y         vg    1.8       y
     13      m     29       y         vg    1.8       y
     11      f     23       y         vg    1.8       y
```

Untuk statistik berikutnya, kita ingin menghubungkan dua kolom tabel. Jadi kita mengekstrak kolom 2 dan 4 dan mengurutkan tabel.

```
>i=sortedrows(MT,[2,4]);  ...
>  writetable(tablecol(MT[i],[2,4])',ctok=[1,2],tok=tok)
```

```
         m         n
         m         n
         m         n
         m         n
         m         n
         m         n
         m         n
         m         y
         m         y
         m         y
         m         y
         m         y
         f         n
         f         y
         f         y
         f         y
         f         y
         f         y
         f         y
         f         y
         f         y
         f         y
         f         y
         f         y
         f         y
```

Dengan getstatistics(), kita juga dapat menghubungkan hitungan 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])
```

```
                 n         y
      m          7         5
      f          1        12
```

Tabel dapat ditulis ke sebuah file.

```
>filename="test.dat"; ...
>writetable(count,labr=tok[xu1],labc=tok[xu2],file=filename);
```

Kemudian kita dapat membaca tabel dari file tersebut.

```
>{MT2,hd,tok2,hdr}=readtable(filename,>clabs,>rlabs); ...
>writetable(MT2,labr=hdr,labc=hd)
```

```
                 n         y
      m          7         5
      f          1        12
```

Dan hapus file tersebut.

```
>fileremove(filename);
```

Dengan plot2d, ada metode yang sangat mudah untuk memplot distribusi data eksperimen.

```
>p=normal(1,1000); //1000 random normal-distributed sample p
>plot2d(p,distribution=20,style="\/"); // plot the random sample p
>plot2d("qnormal(x,0,1)",add=1): // add the standard normal distribution plot
```

Perhatikan perbedaan antara plot batang (sampel) dan kurva normal (distribusi sesungguhnya). Masukkan kembali ketiga perintah tersebut untuk melihat hasil pengambilan sampel yang lain.

Berikut ini adalah perbandingan 10 simulasi dari 1000 nilai terdistribusi normal dengan menggunakan apa yang disebut plot kotak. Plot ini menunjukkan median, kuartil 25% dan 75%, nilai minimal dan maksimal, serta pencilan.

```
>p=normal(10,1000); boxplot(p):
```

Untuk menghasilkan bilangan bulat acak, Euler memiliki intrandom. Mari kita simulasikan pelemparan dadu dan memplot distribusinya.

Kita menggunakan fungsi getmultiplicities(v,x), yang menghitung seberapa sering elemen-elemen dari v muncul di dalam x. Kemudian kita memplot hasilnya menggunakan columnsplot().

```
>k=intrandom(1,6000,6);  ...
>columnsplot(getmultiplicities(1:6,k));  ...
>ygrid(1000,color=red):
```

Meskipun intrandom(n,m,k) menghasilkan bilangan bulat yang terdistribusi secara seragam dari 1 sampai k, adalah mungkin untuk menggunakan distribusi bilangan bulat yang lain dengan randpint().

Pada contoh berikut, probabilitas untuk 1,2,3 adalah 0.4, 0.1, 0.5 secara berurutan.

```
>randpint(1,1000,[0.4,0.1,0.5]); getmultiplicities(1:3,%)
```

```
[378,  102,  520]
```

Euler dapat menghasilkan nilai acak dari lebih banyak distribusi. Lihatlah ke dalam referensi.

Misalnya, kita mencoba distribusi eksponensial. Sebuah variabel acak kontinu X dikatakan memiliki distribusi eksponensial, jika PDF-nya diberikan oleh

$$f_X(x) = \lambda e^{-\lambda x}, \quad x > 0, \quad \lambda > 0,$$

with 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):
```

Untuk banyak distribusi, Euler dapat menghitung fungsi distribusi dan kebalikannya.

```
>plot2d("normaldis",-4,4):
```

Berikut ini 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 untuk berada di area hijau adalah sebagai berikut.

```
>normaldis(5,1,1.5)-normaldis(2,1,1.5)
```

```
0.248662156979
```

Hal ini dapat dihitung secara numerik dengan integral berikut ini.

$$\int_2^5 \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-1}{1.5}\right)^2}\, dx.$$

```
>gauss("qnormal(x,1,1.5)",2,5)
```

```
0.248662156979
```

Mari kita bandingkan distribusi binomial dengan distribusi normal dengan rata-rata dan deviasi yang sama. Fungsi invbindis() menyelesaikan interpolasi linier antara nilai bilangan bulat.

```
>invbindis(0.95,1000,0.5), invnormaldis(0.95,500,0.5*sqrt(1000))
```

```
525.516721219
526.007419394
```

Fungsi qdis() adalah densitas dari distribusi chi-square. Seperti biasa, Euler memetakan vektor ke fungsi ini. Dengan demikian kita mendapatkan plot semua distribusi chi-kuadrat dengan derajat 5 hingga 30 dengan mudah dengan cara berikut.

```
>plot2d("qchidis(x,(5:5:50)')",0,50):
```

Euler memiliki fungsi-fungsi yang akurat untuk mengevaluasi distribusi-distribusi. Mari kita periksa chidis() dengan sebuah integral.

Penamaannya diusahakan untuk konsisten. Sebagai contoh,

- distribusi chi-kuadrat adalah chidis(),
- fungsi kebalikannya adalah invchidis(),
- densitasnya adalah qchidis().

Pelengkap dari distribusi (ekor atas) adalah chicdis().

```
>chidis(1.5,2), integrate("qchidis(x,2)",0,1.5)
```

```
0.527633447259
0.527633447259
```

Untuk menentukan distribusi diskrit Anda sendiri, Anda dapat menggunakan metode berikut.

Pertama, kita tetapkan fungsi distribusinya.

```
>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 yang seragam. Mari kita definisikan sebuah generator bilangan acak untuk ini. Fungsi find(v,x) menemukan nilai x dalam vektor v. Fungsi ini juga dapat digunakan untuk vektor x.

```
>function wrongdice (n,m) := find(wd,random(n,m))
```

Kesalahan ini sangat halus sehingga kita hanya bisa melihatnya setelah melakukan iterasi yang sangat banyak.

```
>columnsplot(getmultiplicities(1:6,wrongdice(1,1000000))):
```

Berikut ini adalah fungsi sederhana untuk memeriksa distribusi seragam dari nilai 1... K dalam v. Kami 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 ini menolak distribusi seragam.

```
>checkrandom(wrongdice(1,1000000))
```

```
0
```

Dan ini menerima generator acak bawaan.

```
>checkrandom(intrandom(1,1000000,6))
```

```
1
```

Kita dapat menghitung distribusi binomial. Pertama, ada binomialsum(), yang mengembalikan probabilitas i atau kurang dari n percobaan.

```
>bindis(410,1000,0.4)
```

```
0.751401349654
```

Fungsi Beta invers digunakan untuk menghitung interval kepercayaan Clopper-Pearson untuk parameter p. Tingkat defaultnya adalah alpha.

Arti dari interval ini adalah jika p berada di luar interval, hasil yang diamati sebesar 410 dalam 1000 jarang terjadi.

```
>clopperpearson(410,1000)
```

```
[0.37932,  0.441212]
```

Perintah berikut ini adalah cara langsung untuk mendapatkan hasil di atas. Tetapi untuk 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
```

Omong-omong, invbinsum() menghitung kebalikan dari binomialsum().

```
>invbindis(0.75,1000,0.4)
```

```
409.932733047
```

Dalam Bridge, kita mengasumsikan 5 kartu yang terbuka (dari 52 kartu) di dua tangan (26 kartu). Mari kita hitung probabilitas distribusi yang lebih buruk dari 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])
```

```
        381        100        519
        376         91        533
        417         80        503
        440         94        466
        406        112        482
        408         94        498
        395        107        498
```

399	96	505
428	87	485
400	99	501

Untuk memplot data, kami mencoba hasil pemilihan umum Jerman sejak tahun 1990, yang 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 pesta, kami menggunakan serangkaian nama.

```
>P:=["CDU/CSU","SPD","FDP","Gr","Li"];
```

Mari kita cetak persentasenya dengan baik.

Pertama kita ekstrak kolom-kolom yang diperlukan. Kolom 3 sampai 7 adalah kursi masing-masing partai, dan kolom 2 adalah jumlah total kursi. kolom adalah tahun pemilihan.

```
>BT:=BW[,3:7]; BT:=BT/sum(BT); YT:=BW[,1]';
```

Kemudian kita mencetak statistik dalam bentuk tabel. Kita menggunakan nama sebagai judul kolom, dan tahun sebagai judul baris. Lebar default untuk kolom adalah wc = 10, tetapi kami lebih suka output yang lebih padat. Kolom-kolom akan diperluas untuk label-label kolom, jika perlu.

```
>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 ini mengekstrak jumlah persentase dua partai besar yang menunjukkan bahwa partai-partai kecil telah memperoleh suara 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. Alternatif lainnya adalah memanggil plot2d dua kali dengan >add.

```
>statplot(YT,BT1,"b"):
```

Tentukan beberapa warna untuk masing-masing pihak.

```
>CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.8,0,0)];
```

Sekarang kita dapat memplot hasil pemilu 2009 dan perubahannya ke dalam satu plot menggunakan figure. Kita dapat menambahkan vektor kolom pada 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 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 deretan data statistik dalam bentuk kolom. Kami menyediakan label untuk baris dan kolom. angle adalah sudut pandang.

```
>columnsplot3d(BT,scols=P,srows=YT, ...
>  angle=30°,ccols=CP):
```

Representasi lainnya adalah plot mosaik. Perhatikan bahwa kolom-kolom pada plot mewakili kolom-kolom pada matriks di sini. Karena panjangnya label CDU/CSU, kita mengambil jendela yang lebih kecil dari biasanya.

```
>shrinkwindow(>smaller);  ...
>mosaicplot(BT',srows=YT,scols=P,color=CP,style="#"); ...
>shrinkwindow():
```

Kita juga bisa membuat diagram lingkaran. Karena warna hitam dan kuning membentuk sebuah koalisi, kita menyusun ulang elemen-elemennya.

```
>i=[1,3,5,4,2]; piechart(BW[6,3:7][i],color=CP[i],lab=P[i]):
```

Berikut ini jenis plot yang lain.

```
>starplot(normal(1,10)+4,lab=1:10,>rays):
```

Beberapa plot di plot2d bagus untuk statika. Berikut ini adalah plot impuls dari data acak, yang terdistribusi secara seragam dalam [0,1].

```
>plot2d(makeimpulse(1:10,random(1,10)),>bar):
```

Tetapi untuk data yang terdistribusi secara eksponensial, kita mungkin memerlukan plot logaritmik.

```
>logimpulseplot(1:10,-log(random(1,10))*10):
```

Fungsi columnsplot() lebih mudah digunakan, karena hanya membutuhkan sebuah vektor nilai. Selain itu, fungsi ini dapat mengatur labelnya menjadi apa pun yang kita inginkan, kita telah mendemonstrasikan hal ini dalam tutorial ini.

Berikut ini adalah aplikasi lain, di mana kita menghitung karakter dalam sebuah kalimat dan memplot statistik.

```
>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):
```

Anda juga dapat menetapkan sumbu secara manual.

```
>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 ini adalah cara untuk memplot frekuensi angka dalam vektor.

Kami membuat vektor angka acak bilangan bulat 1 hingga 6.

```
>v:=intrandom(1,10,10)
```

```
[8,  5,  8,  8,  6,  8,  8,  3,  5,  5]
```

Kemudian ekstrak nomor unik dalam v.

```
>vu:=unique(v)
```

```
[3,  5,  6,  8]
```

Dan memplot frekuensi dalam plot kolom.

```
>columnsplot(getmultiplicities(vu,v),lab=vu,style="/"):
```

Kami ingin mendemonstrasikan fungsi untuk distribusi nilai empiris.

```
>x=normal(1,20);
```

Fungsi empdist(x,vs) membutuhkan larik nilai yang telah diurutkan. Jadi kita harus mengurutkan x sebelum dapat menggunakannya.

```
>xs=sort(x);
```

Kemudian kita memplot distribusi empiris dan beberapa batang kepadatan ke dalam satu plot. Alih-alih plot batang untuk distribusi, kali ini kami menggunakan plot gigi gergaji.

```
>figure(2,1); ...
>figure(1); plot2d("empdist",-4,4;xs); ...
>figure(2); plot2d(histo(x,v=-4:0.2:4,<bar));  ...
>figure(0):
```

Plot sebaran mudah dilakukan di Euler dengan plot titik biasa. Grafik berikut ini menunjukkan bahwa X dan X+Y berkorelasi positif secara jelas.

```
>x=normal(1,100); plot2d(x,x+rotright(x),>points,style=".."):
```

Sering kali, kita ingin membandingkan dua sampel dari distribusi yang berbeda. Hal ini dapat dilakukan dengan plot kuantil-kuantil.

Untuk pengujian, kami mencoba distribusi student-t 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):
```

Plot ini dengan jelas menunjukkan bahwa nilai yang terdistribusi normal cenderung lebih kecil pada ujung yang ekstrim.

Jika kita memiliki dua distribusi dengan ukuran yang berbeda, kita dapat memperluas distribusi yang lebih kecil atau memperkecil distribusi yang lebih besar. Fungsi berikut ini bagus untuk keduanya. Fungsi 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 linier dapat dilakukan dengan fungsi polyfit() atau berbagai fungsi kecocokan.

Sebagai permulaan, kita mencari 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
```

Kami ingin membandingkan kecocokan tanpa bobot dan dengan bobot. Pertama, koefisien dari kecocokan linier.

```
>p=polyfit(x,y,1)
```

```
[0.733333,  0.812121]
```

Sekarang, koefisien dengan bobot yang menekankan nilai terakhir.

```
>w &= "exp(-(x-10)^2/10)"; pw=polyfit(x,y,1,w=w(x))
```

```
[4.71566,  0.38319]
```

Kami menempatkan semuanya ke dalam satu plot untuk titik-titik dan 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 survei tentang siswa, usia mereka, usia orang tua mereka, dan jumlah saudara kandung dari sebuah file.

Tabel ini berisi "m" dan "f" pada kolom kedua. Kita menggunakan variabel tok2 untuk mengatur terjemahan yang tepat dan bukannya membiarkan readtable() mengumpulkan terjemahan.

```
>{MS,hd}:=readtable("table1.dat",tok2:=["m","f"]);  ...
>writetable(MS,labc=hd,tok2:=["m","f"]);
```

Person	Sex	Age	Mother	Father	Siblings
1	m	29	58	61	1
2	f	26	53	54	2
3	m	24	49	55	1
4	f	25	56	63	3
5	f	25	49	53	0
6	f	23	55	55	2
7	m	23	48	54	2
8	m	27	56	58	1
9	m	25	57	59	1
10	m	24	50	54	1
11	f	26	61	65	1
12	m	24	50	52	1
13	m	29	54	56	1
14	m	28	48	51	2
15	f	23	52	52	1
16	m	24	45	57	1
17	f	24	59	63	0
18	f	23	52	55	1
19	m	24	54	61	2
20	f	23	54	55	1

Bagaimana usia saling bergantung satu sama lain? Kesan pertama datang dari scatterplot berpasangan.

```
>scatterplots(tablecol(MS,3:5),hd[3:5]):
```

Jelas bahwa usia ayah dan ibu saling bergantung satu sama lain. Mari kita tentukan dan plot garis regresinya.

```
>cs:=MS[,4:5]'; ps:=polyfit(cs[1],cs[2],1)
```

```
[17.3789,  0.740964]
```

Ini jelas merupakan model yang salah. Garis regresinya 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 terlalu banyak.

Sebaliknya, kita menduga fungsi seperti s = a + t. Kemudian a adalah rata-rata dari s-t. Ini adalah perbedaan usia rata-rata antara ayah dan ibu.

```
>da:=mean(cs[2]-cs[1])
```

```
3.65
```

Mari kita plotkan ini ke dalam satu scatter plot.

```
>plot2d(cs[1],cs[2],>points);  ...
>plot2d("evalpoly(x,ps)",color=red,style=".",>add);  ...
>plot2d("x+da",color=blue,>add):
```

Berikut ini adalah plot kotak dari kedua usia tersebut. Ini hanya menunjukkan, bahwa usia keduanya berbeda.

```
>boxplot(cs,["mothers","fathers"]):
```

Sangat menarik bahwa perbedaan dalam median tidak sebesar perbedaan dalam mean.

```
>median(cs[2])-median(cs[1])
```

```
1.5
```

Koefisien korelasi menunjukkan korelasi positif.

```
>correl(cs[1],cs[2])
```

```
0.7588307236
```

Korelasi peringkat adalah ukuran untuk urutan yang sama dalam kedua vektor. Korelasi ini juga cukup positif.

```
>rankcorrel(cs[1],cs[2])
```

```
0.758925292358
```

Tentu saja, bahasa EMT dapat digunakan untuk memprogram fungsi baru. Misalnya, kita mendefinisikan fungsi kemiringan.

$$\mathrm{sk}(x) = \frac{\sqrt{n}\sum_i (x_i - m)^3}{\left(\sum_i (x_i - m)^2\right)^{3/2}}$$

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 dapat dengan mudah menggunakan bahasa matriks untuk mendapatkan implementasi yang sangat singkat dan efisien. Mari kita coba fungsi ini.

```
>data=normal(20); skew(normal(10))
```

```
-0.198710316203
```

Berikut ini adalah fungsi lain, yang disebut koefisien kemencengan Pearson.

```
>function skew1 (x) := 3*(mean(x)-median(x))/dev(x)
>skew1(data)
```

```
-0.0801873249135
```

Euler dapat digunakan untuk mensimulasikan kejadian acak. Kita telah melihat contoh sederhana di atas. Berikut ini adalah contoh lainnya, yang mensimulasikan 1000 kali pelemparan 3 dadu, dan menanyakan distribusi dari jumlah tersebut.

```
>ds:=sum(intrandom(1000,3,6))'; fs=getmultiplicities(3:18,ds)
```

```
 [5,  17,  35,  44,  75,  97,  114,  116,  143,  116,  104,  53,  40,
 22,  13,  6]
```

Kita bisa merencanakan ini sekarang.

```
>columnsplot(fs,lab=3:18):
```

Untuk menentukan distribusi yang diharapkan tidaklah mudah. Kami menggunakan rekursi tingkat lanjut untuk hal ini.

Fungsi berikut ini menghitung jumlah cara angka k dapat direpresentasikan sebagai jumlah n angka dalam rentang 1 hingga m. Fungsi 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 ini adalah hasil dari 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]
```

Kami menambahkan nilai yang diharapkan ke 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 berdistribusi normal 0-1 adalah 1/sqrt(n).

```
>longformat; 1/sqrt(10)
```

```
0.316227766017
```

Mari kita periksa hal ini dengan sebuah simulasi. Kami menghasilkan 10.000 kali 10 vektor acak.

```
>M=normal(10000,10); dev(mean(M)')
```

```
0.319493614817
```

```
>plot2d(mean(M)',>distribution):
```

Median dari 10 bilangan acak berdistribusi normal 0-1 memiliki deviasi yang lebih besar.

```
>dev(median(M)')
```

```
0.374460271535
```

Karena kita dapat dengan mudah menghasilkan jalan acak, kita dapat mensimulasikan proses Wiener. Kami mengambil 1000 langkah dari 1000 proses. Kami kemudian memplot deviasi standar dan rata-rata dari langkah ke-n dari proses-proses ini bersama dengan 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):
```

Tes adalah alat yang penting dalam statistik. Dalam Euler, banyak tes yang diterapkan. Semua tes ini mengembalikan kesalahan yang kita terima jika kita menolak hipotesis nol.

Sebagai contoh, kita menguji lemparan dadu untuk distribusi yang seragam. Pada 600 lemparan, kita mendapatkan nilai berikut, yang kita masukkan ke dalam uji chi-kuadrat.

```
>chitest([90,103,114,101,103,89],dup(100,6)')
```

```
0.498830517952
```

Uji chi-square 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)
```

```
0.526
```

Kesalahan ini terlalu besar. Jadi kita tidak bisa menolak distribusi seragam. Ini tidak membuktikan bahwa dadu kita adil. Tetapi kita tidak dapat menolak hipotesis kita.

Selanjutnya kita buat 1000 lemparan dadu dengan menggunakan generator bilangan acak, dan lakukan pengujian yang sama.

```
>n=1000; t=random([1,n*6]); chitest(count(t*6,6),dup(n,6)')
```

```
0.528028118442
```

Mari kita uji nilai rata-rata 100 dengan uji-t.

```
>s=200+normal([1,100])*10; ...
>ttest(mean(s),dev(s),100,200)
```

```
0.0218365848476
```

The function ttest() needs the mean value, the deviation, the number of data, and the mean value to test for.

Now let us check two measurements for the same mean. We reject the hypothesis that they have the same mean, if the result is <0.05.

```
>tcomparedata(normal(1,10),normal(1,10))
```

```
0.38722000942
```

Jika kita menambahkan bias pada satu distribusi, kita akan mendapatkan lebih banyak penolakan. Ulangi simulasi ini beberapa kali untuk melihat efeknya.

```
>tcomparedata(normal(1,10),normal(1,10)+2)
```

```
5.60009101758e-07
```

Pada contoh berikut, kita membuat 20 lemparan dadu secara acak sebanyak 100 kali dan menghitung jumlah dadu yang muncul. Rata-rata harus ada $20/6 = 3{,}3$ mata dadu.

```
>R=random(100,20); R=sum(R*6<=1)'; mean(R)
```

```
3.28
```

Sekarang kita bandingkan jumlah satu dengan distribusi binomial. Pertama, kita memplot distribusi angka satu.

```
>plot2d(R,distribution=max(R)+1,even=1,style="\/"):
>t=count(R,21);
```

Kemudian kami menghitung nilai yang diharapkan.

```
>n=0:20; b=bin(20,n)*(1/6)^n*(5/6)^(20-n)*100;
```

Kami harus mengumpulkan beberapa angka untuk mendapatkan 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-square menolak hipotesis bahwa distribusi kita adalah distribusi binomial, jika hasilnya <0,05.

```
>chitest(t1,b1)
```

```
0.53921579764
```

Contoh berikut ini berisi hasil dari dua kelompok orang (laki-laki dan perempuan, katakanlah) yang memberikan suara untuk 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
```

Kami ingin menguji independensi suara dari jenis kelamin. Uji tabel chi^2 melakukan hal ini. Hasilnya terlalu besar untuk menolak independensi. Jadi kita tidak dapat mengatakan, jika pemungutan suara tergantung pada jenis kelamin dari data ini.

```
>tabletest(A)
```

```
0.990701632326
```

Berikut ini adalah tabel yang diharapkan, jika kita mengasumsikan frekuensi pemungutan suara yang diamati.

```
>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 terkoreksi. Karena koefisien ini sangat dekat dengan 0, kami menyimpulkan bahwa pemungutan suara tidak bergantung pada jenis kelamin.

```
>contingency(A)
```

```
0.0427225484717
```

Selanjutnya kita menggunakan analisis varians (uji F) untuk menguji tiga sampel data yang terdistribusi secara normal dengan nilai rata-rata yang sama. Metode ini disebut ANOVA (analisis varians). Dalam Euler, fungsi varanalysis() digunakan.

```
>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
```

Ini berarti, kami menolak hipotesis nilai rata-rata yang sama. Kami melakukan ini 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 gabungan.

```
>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 tentang kesetaraan adalah uji peringkat. Uji ini jauh lebih tajam daripada uji median.

```
>ranktest(a,b)
```

```
0.00199969612469
```

Dalam contoh berikut ini, kedua distribusi memiliki rata-rata yang sama.

```
>ranktest(random(1,100),random(1,50)*3-1)
```

```
0.129608141484
```

Sekarang mari kita coba mensimulasikan dua perawatan a dan b yang diterapkan pada 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 memutuskan, apakah a lebih baik daripada b.

```
>signtest(a,b)
```

```
0.0546875
```

Ini adalah kesalahan yang terlalu besar. Kita tidak dapat 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 coba dua pengujian lagi dengan menggunakan rangkaian yang dihasilkan.

```
>wilcoxon(normal(1,20),normal(1,20)-1)
```

```
0.0068706451766
```

```
>wilcoxon(normal(1,20),normal(1,20))
```

```
0.275145971064
```

Berikut ini adalah tes untuk generator bilangan acak. Euler menggunakan generator yang sangat bagus, jadi kita tidak perlu mengharapkan adanya masalah.

Pertama, kita akan membangkitkan sepuluh juta bilangan acak dalam [0,1].

```
>n:=10000000; r:=random(1,n);
```

Selanjutnya, kami menghitung jarak antara dua angka yang kurang dari 0,05.

```
>a:=0.05; d:=differences(nonzeros(r<a));
```

Terakhir, kami memplot berapa kali, setiap jarak yang 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):
```

Menghapus data.

```
>remvalue n;
```

Pengantar untuk Pengguna Proyek R

Jelas, EMT tidak bersaing dengan R sebagai sebuah paket statistik. Namun, ada banyak prosedur dan fungsi statistik yang tersedia di EMT juga. Jadi EMT dapat memenuhi kebutuhan dasar. Bagaimanapun, EMT hadir dengan paket numerik dan sistem aljabar komputer.

Buku ini diperuntukkan bagi Anda yang sudah terbiasa dengan R, tetapi perlu mengetahui perbedaan sintaks EMT dan R. Kami mencoba memberikan gambaran umum mengenai hal-hal yang jelas dan kurang jelas yang perlu Anda ketahui.

Selain itu, kami juga membahas cara-cara untuk bertukar data di antara kedua sistem tersebut.

Perhatikan bahwa ini adalah pekerjaan yang sedang berlangsung.

Sintaks Dasar

Hal pertama yang Anda pelajari dalam R adalah membuat sebuah vektor. Dalam EMT, perbedaan utamanya adalah bahwa operator : dapat mengambil ukuran langkah. Selain itu, operator ini memiliki daya ikat 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. Anda dapat menggunakan vektor untuk menggabungkan beberapa hal.

Contoh berikut ini, seperti banyak contoh lainnya, berasal dari "Interoduksi ke R" yang disertakan dengan proyek R. Jika Anda membaca PDF ini, Anda akan menemukan bahwa saya mengikuti alurnya 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 dengan ukuran langkah EMT digantikan oleh fungsi seq() dalam R. Kita dapat menulis fungsi ini dalam 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() dari R tidak ada dalam EMT. Untuk input vektor, dapat dituliskan 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]
```

Perhatikan bahwa "=" atau ":=" digunakan untuk penugasan. Operator "->" digunakan untuk unit dalam EMT.

```
>125km -> " miles"
```

```
77.6713990297 miles
```

Operator "<-" untuk penugasan menyesatkan, dan bukan ide yang baik untuk R. Berikut ini akan membandingkan a dan -4 dalam EMT.

```
>a=2; a<-4
```

```
0
```

Dalam R, "a<-4<3" bisa digunakan, tetapi "a<-4<-3" tidak. Saya juga mengalami ambiguitas yang sama di EMT, tetapi saya mencoba untuk menghilangkannya.

EMT dan R memiliki vektor dengan tipe boolean. Tetapi dalam EMT, angka 0 dan 1 digunakan untuk merepresentasikan salah dan benar. Dalam R, nilai benar dan salah tetap dapat digunakan dalam aritmatika biasa seperti dalam EMT.

```
>x<5, %*x
```

```
[0,  0,  1,  0,  0]
[0,  0,  3.1,  0,  0]
```

EMT melempar kesalahan atau menghasilkan NAN tergantung pada flag "kesalahan".

```
>errors off; 0/0, isNAN(sqrt(-1)), errors on;
```

```
NAN
1
```

String sama saja dalam R dan EMT. Keduanya berada di lokal saat ini, bukan di Unicode.

Dalam R ada paket-paket untuk Unicode. Dalam EMT, sebuah string dapat berupa string Unicode. Sebuah string Unicode dapat diterjemahkan ke pengkodean lokal dan sebaliknya. Selain itu, u"..." dapat berisi entitas HTML.

```
>u"&#169; Ren&eacut; Grothmann"
```

```
© René Grothmann
```

Berikut ini mungkin atau mungkin tidak ditampilkan dengan benar pada sistem Anda sebagai A dengan titik dan tanda hubung di atasnya. Hal ini tergantung pada jenis huruf yang Anda gunakan.

```
>chartoutf([480])
```

Penggabungan string dilakukan dengan "+" atau "|". Ini dapat menyertakan angka, yang akan dicetak dalam format saat ini.

```
>"pi = "+pi
```

```
pi = 3.14159265359
```

Sebagian besar waktu, ini akan bekerja seperti pada R.

Tetapi EMT akan menginterpretasikan indeks negatif dari bagian 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 drop().

```
>drop(x,2)
```

```
[10.4,  3.1,  6.4,  21.7]
```

Vektor logika tidak diperlakukan secara berbeda dengan indeks di EMT, berbeda dengan R. Anda harus mengekstrak 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 di R, vektor indeks dapat berisi pengulangan.

```
>x[[1,2,2,1]]
```

```
[10.4,  5.6,  5.6,  10.4]
```

Namun pemberian nama untuk indeks tidak dimungkinkan dalam EMT. Untuk paket statistik, hal ini mungkin sering diperlukan untuk memudahkan akses ke elemen-elemen vektor.

Untuk meniru perilaku ini, kita dapat mendefinisikan sebuah fungsi sebagai 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)]; ... ...
             ^

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]
```

EMT memiliki lebih banyak tipe data yang tetap dibandingkan R. Jelas, dalam R terdapat vektor yang berkembang. Anda bisa mengatur sebuah vektor numerik kosong v dan memberikan sebuah nilai pada elemen v[17]. Hal ini tidak mungkin dilakukan dalam EMT.

Hal berikut ini sedikit tidak efisien.

```
>v=[]; for i=1 to 10000; v=v|i; end;
```

EMT sekarang akan membuat vektor dengan v dan i yang ditambahkan pada tumpukan dan menyalin vektor tersebut kembali ke variabel global v.

Semakin efisien mendefinisikan vektor terlebih dahulu.

```
>v=zeros(10000); for i=1 to 10000; v[i]=i; end;
```

Untuk mengubah jenis tanggal di EMT, Anda dapat menggunakan fungsi seperti complex().

```
>complex(1:4)
```

```
 [ 1+0i ,  2+0i ,  3+0i ,  4+0i  ]
```

Konversi ke string hanya dapat dilakukan untuk tipe data dasar. Format saat ini digunakan untuk penggabungan string sederhana. Tetapi ada fungsi-fungsi seperti print() atau frac().

Untuk vektor, Anda dapat dengan mudah menulis fungsi Anda 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 komunikasi dengan Maxima, ada sebuah fungsi convertmxm(), yang juga dapat digunakan untuk memformat vektor untuk output.

```
>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

Pada pengantar R terdapat sebuah contoh dengan apa yang disebut faktor.

Berikut ini 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"];
```

Asumsikan, 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 tersebut. Sebagai sebuah program statistik, R memiliki fungsi factor() dan tappy() untuk hal ini.

EMT dapat melakukan hal ini dengan mencari indeks dari wilayah-wilayah di dalam daftar unik dari wilayah-wilayah tersebut.

```
>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 titik ini, kita dapat menulis fungsi perulangan kita sendiri untuk melakukan berbagai hal 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 sedikit tidak efisien, karena menghitung wilayah unik untuk setiap i, tetapi berfungsi.

```
>tappl(auterr,"mean",austates,incomes)
```

```
[44.5,  57.3333333333,  55.5,  53.6,  55,  60.5,  56,  52.25]
```

Perhatikan bahwa ini bekerja untuk setiap vektor wilayah.

```
>tappl(["act","nsw"],"mean",austates,incomes)
```

```
[44.5,  57.3333333333]
```

Sekarang, paket statistik EMT mendefinisikan tabel seperti halnya di R. Fungsi readtable() dan writetable() dapat digunakan untuk input dan output.

Jadi kita dapat mencetak rata-rata pendapatan negara di 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 tersebut harus disimpan dengan jelas dalam sebuah koleksi dengan jenis dan kategorinya (negara bagian dan wilayah dalam contoh kita). Untuk EMT, kita menambahkan 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 ini 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 dapat meniru tapply() dengan cara berikut. Ini akan mengembalikan sebuah tabel sebagai kumpulan 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 kategori (faktor) yang tidak memiliki data. Tetapi kita harus memeriksa panjang t yang benar dan kebenaran koleksi tf.

Tabel ini bisa dicetak sebagai sebuah 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
```

EMT hanya memiliki dua dimensi untuk array. Tipe datanya disebut matriks. Akan lebih mudah untuk menulis fungsi untuk dimensi yang lebih tinggi atau pustaka C untuk ini.

R memiliki lebih dari dua dimensi. Dalam R, larik adalah sebuah vektor dengan sebuah bidang dimensi.

Dalam EMT, sebuah vektor adalah sebuah matriks dengan satu baris. Ini bisa dibuat menjadi sebuah 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
```

Ekstraksi baris dan kolom, atau sub-matriks, sama seperti di R.

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

```
          2           3
          7           8
         12          13
         17          18
```

Namun, dalam R dimungkinkan untuk mengatur daftar indeks tertentu dari vektor ke suatu nilai. Hal yang sama juga dapat dilakukan dalam EMT hanya dengan sebuah perulangan.

```
>function setmatrixvalue (M, i, j, v) ...
```

```
loop 1 to max(length(i),length(j),length(v))
   M[i{#},j{#}] = v{#};
end;
endfunction
```

Kami mendemonstrasikan hal ini untuk menunjukkan bahwa matriks diteruskan dengan referensi dalam 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
```

Hasil kali luar dalam EMT hanya dapat dilakukan di antara vektor. Hal ini otomatis karena bahasa matriks. Satu vektor harus berupa vektor kolom dan 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
```

Dalam pengantar PDF untuk R ada sebuah contoh, yang menghitung distribusi ab-cd untuk a, b, c, d yang dipilih dari 0 sampai n secara acak. Solusinya dalam R adalah membentuk sebuah matriks 4 dimensi dan menjalankan table() di atasnya.

Tentu saja, ini bisa dicapai dengan sebuah perulangan. Tetapi perulangan tidak efektif dalam EMT atau R. Dalam EMT, kita bisa menulis perulangan dalam C dan itu adalah solusi tercepat.

Tetapi kita ingin meniru perilaku R. Untuk ini, kita perlu meratakan perkalian ab dan membuat sebuah 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 kelipatan yang tepat, EMT dapat menghitung frekuensi dalam vektor.

```
>getfrequencies(q,-50:10:50)
```

```
[0,  23,  132,  316,  602,  801,  333,  141,  53,  0]
```

Cara yang paling mudah untuk memplot ini sebagai distribusi adalah sebagai berikut.

```
>plot2d(q,distribution=11):
```

Tetapi juga memungkinkan untuk menghitung jumlah dalam interval yang dipilih sebelumnya. Tentu saja, berikut ini menggunakan getfrequencies() secara internal.

Karena fungsi histo() mengembalikan frekuensi, kita perlu menskalakannya sehingga integral di bawah grafik batang adalah 1.

```
>{x,y}=histo(q,v=-55:10:55); y=y/sum(y)/differences(x); ...
>plot2d(x,y,>bar,style="/"):
```

EMT memiliki dua jenis daftar. Pertama adalah daftar global yang dapat diubah, dan yang kedua adalah jenis daftar yang tidak dapat diubah. Kita tidak peduli dengan daftar global di sini.

Tipe daftar yang tidak dapat diubah disebut koleksi dalam EMT. Ia berperilaku seperti struktur dalam C, tetapi elemen-elemennya hanya diberi nomor dan tidak diberi nama.

```
>L={{"Fred","Flintstone",40,[1990,1992]}}
```

```
Fred
Flintstone
40
[1990,  1992]
```

Saat ini elemen-elemen tersebut tidak memiliki nama, meskipun nama dapat ditetapkan untuk tujuan khusus. Elemen-elemen tersebut diakses dengan angka.

```
>(L[4])[2]
```

```
1992
```

Input dan Output File (Membaca dan Menulis Data)

Anda mungkin sering ingin mengimpor matriks data dari sumber lain ke EMT. Tutorial ini akan menjelaskan kepada Anda tentang berbagai cara untuk melakukan hal tersebut. Fungsi yang sederhana adalah writematrix() dan readmatrix().

Mari kita tunjukkan bagaimana cara membaca dan menulis sebuah vektor real ke sebuah file.

```
>a=random(1,100); mean(a), dev(a),
```

```
0.49815
0.28037
```

Untuk menulis data ke sebuah berkas, kita menggunakan fungsi writematrix().

Karena pengenalan ini kemungkinan besar berada di sebuah direktori, di mana pengguna tidak memiliki akses tulis, kita menulis data ke direktori home pengguna. Untuk notebook sendiri, hal ini tidak diperlukan, karena file data akan ditulis ke dalam direktori yang sama.

```
>filename="test.dat";
```

Sekarang kita tuliskan vektor kolom a' ke dalam file. Hal ini akan menghasilkan satu angka pada setiap baris file.

```
>writematrix(a',filename);
```

Untuk membaca data, kita menggunakan readmatrix().

```
>a=readmatrix(filename)';
```

Dan hapus file tersebut.

```
>fileremove(filename);
>mean(a), dev(a),
```

```
0.49815
0.28037
```

Fungsi writematrix() atau writetable() dapat dikonfigurasi untuk bahasa lain.

Sebagai contoh, jika Anda memiliki sistem bahasa Indonesia (titik desimal dengan koma), Excel Anda membutuhkan nilai dengan koma desimal yang dipisahkan oleh titik koma dalam file csv (defaultnya adalah nilai yang dipisahkan dengan koma). File "test.csv" berikut ini akan muncul di folder cuurent Anda.

```
>filename="test.csv"; ...
>writematrix(random(5,3),file=filename,separator=",");
```

Anda sekarang dapat membuka file ini dengan Excel Indonesia secara langsung.

```
>fileremove(filename);
```

Terkadang kita memiliki string dengan token seperti berikut ini.

```
>s1:="f m m f m m m f f f m m f";  ...
>s2:="f f f m m f f";
```

Untuk menandai ini, kita mendefinisikan vektor token.

```
>tok:=["f","m"]
```

```
 f
 m
```

Kemudian kita dapat menghitung berapa kali setiap token muncul dalam string, dan memasukkan hasilnya ke dalam tabel.

```
>M:=getmultiplicities(tok,strtokens(s1))_ ...
>  getmultiplicities(tok,strtokens(s2));
```

Tulis tabel dengan tajuk token.

```
>writetable(M,labc=tok,labr=1:2,wc=8)
```

```
               f       m
       1       6       7
       2       5       2
```

Untuk statika, EMT dapat membaca dan menulis tabel.

```
>file="test.dat"; open(file,"w"); ...
>writeln("A,B,C"); writematrix(random(3,3)); ...
>close();
```

File terlihat seperti ini.

```
>printfile(file)
```

```
A,B,C
0.7003664386138074,0.1875530821001213,0.3262339279660414
0.5926249243193858,0.1522927283984059,0.368140583062521
0.8065535209872989,0.7265910840408142,0.7332619844597152
```

Fungsi readtable() dalam bentuknya yang paling sederhana dapat membaca ini dan mengembalikan sebuah koleksi nilai dan baris judul.

```
>L=readtable(file,>list);
```

Koleksi ini dapat dicetak dengan writetable() ke buku catatan, atau ke sebuah file.

```
>writetable(L,wc=10,dc=5)
```

```
         A         B         C
   0.70037   0.18755   0.32623
   0.59262   0.15229   0.36814
   0.80655   0.72659   0.73326
```

Matriks nilai adalah elemen pertama dari L. Perhatikan bahwa mean() dalam EMT menghitung nilai rata-rata dari baris-baris matriks.

```
>mean(L[1])
```

```
0.40472
0.37102
0.75547
```

Pertama, mari kita tulis sebuah matriks ke dalam sebuah file. Untuk keluarannya, kami membuat file di direktori kerja saat ini.

```
>file="test.csv";  ...
>M=random(3,3); writematrix(M,file);
```

Berikut ini adalah isi file ini.

```
>printfile(file)
```

```
0.8221197733097619,0.821531098722547,0.7771240608094004
0.8482947121863489,0.3237767724883862,0.6501422353377985
0.1482301827518109,0.3297459716109594,0.6261901074210923
```

CVS ini dapat dibuka di sistem bahasa Inggris ke Excel dengan klik dua kali. Jika Anda mendapatkan file seperti itu pada sistem Jerman, Anda perlu mengimpor data ke Excel dengan memperhatikan titik desimal.

Namun, titik desimal juga merupakan format default untuk EMT. Anda dapat membaca sebuah matriks dari sebuah file dengan readmatrix().

```
>readmatrix(file)
```

```
    0.82212   0.82153   0.77712
    0.84829   0.32378   0.65014
    0.14823   0.32975   0.62619
```

Dimungkinkan untuk menulis beberapa matriks ke dalam satu file. Perintah open() dapat membuka file untuk menulis dengan parameter "w". Standarnya adalah "r" untuk membaca.

```
>open(file,"w"); writematrix(M); writematrix(M'); close();
```

Matriks-matriks tersebut dipisahkan oleh sebuah baris kosong. Untuk membaca matriks, buka file dan panggil 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 matriks sebagai CSV (nilai yang dipisahkan dengan koma). Pada Excel 2007, gunakan "save as" dan "format lain", lalu pilih "CSV". Pastikan, tabel saat ini hanya berisi data yang ingin Anda ekspor.

Berikut ini adalah contohnya.

```
>printfile("excel-data.csv")
```

```
0;1000;1000
1;1051,271096;1072,508181
2;1105,170918;1150,273799
3;1161,834243;1233,67806
4;1221,402758;1323,129812
5;1284,025417;1419,067549
6;1349,858808;1521,961556
7;1419,067549;1632,31622
8;1491,824698;1750,6725
9;1568,312185;1877,610579
10;1648,721271;2013,752707
```

Seperti yang Anda lihat, sistem Jerman saya menggunakan titik koma sebagai pemisah dan koma desimal. Anda dapat mengubahnya di pengaturan sistem atau di Excel, tetapi tidak perlu untuk membaca matriks ke dalam EMT.

Cara termudah untuk membaca ini ke dalam Euler adalah readmatrix(). Semua koma digantikan oleh titik dengan parameter >comma. Untuk CSV bahasa Inggris, hilangkan saja parameter ini.

```
>M=readmatrix("excel-data.csv",>comma)
```

```
   0      1000      1000
   1    1051.3    1072.5
   2    1105.2    1150.3
   3    1161.8    1233.7
   4    1221.4    1323.1
   5      1284    1419.1
   6    1349.9      1522
   7    1419.1    1632.3
   8    1491.8    1750.7
   9    1568.3    1877.6
  10    1648.7    2013.8
```

Mari kita rencanakan ini.

```
>plot2d(M'[1],M'[2:3],>points,color=[red,green]'):
```

Ada beberapa cara yang lebih mendasar untuk membaca data dari file. Anda dapat membuka file dan membaca angka baris demi baris. Fungsi getvectorline() akan membaca angka dari sebuah baris data. Secara default, fungsi ini mengharapkan sebuah titik desimal. Tetapi fungsi ini juga dapat menggunakan koma desimal, jika Anda memanggil setdecimaldot(",") sebelum menggunakan fungsi ini.

Fungsi berikut ini adalah contohnya. 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)
```

```
0.82212   0.82153   0.77712
0.84829   0.32378   0.65014
0.14823   0.32975   0.62619
```

Anda juga dapat membaca semua angka dalam file tersebut dengan getvector().

```
>open(file); v=getvector(10000); close(); redim(v[1:9],3,3)
```

```
0.82212   0.82153   0.77712
0.84829   0.32378   0.65014
0.14823   0.32975   0.62619
```

Dengan demikian, sangat mudah untuk menyimpan vektor nilai, satu nilai di setiap baris dan membaca kembali vektor ini.

```
>v=random(1000); mean(v)
```

```
0.50303
```

```
>writematrix(v',file); mean(readmatrix(file)')
```

```
0.50303
```

Tabel dapat digunakan untuk membaca atau menulis data numerik. Sebagai contoh, kita menulis tabel dengan judul baris dan kolom ke file.

```
>file="test.tab"; M=random(3,3);  ...
>open(file,"w");  ...
>writetable(M,separator=",",labc=["one","two","three"]);  ...
>close(); ...
>printfile(file)
```

```
 one,two,three
       0.09,       0.39,       0.86
       0.39,       0.86,       0.71
        0.2,       0.02,       0.83
```

File ini dapat diimpor ke Excel.

Untuk membaca file di EMT, kita menggunakan readtable().

```
>{M,headings}=readtable(file,>clabs); ...
>writetable(M,labc=headings)
```

```
        one       two     three
       0.09      0.39      0.86
       0.39      0.86      0.71
        0.2      0.02      0.83
```

Anda bahkan dapat mengevaluasi setiap baris dengan tangan. Misalkan, kita memiliki baris dengan format berikut.

```
>line="2020-11-03,Tue,1'114.05"
```

```
2020-11-03,Tue,1'114.05
```

First we can tokenize the line.

```
>vt=strtokens(line)
```

```
2020-11-03
Tue
1'114.05
```

Kemudian, kita dapat mengevaluasi setiap elemen garis dengan 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 ekspresi reguler, Anda dapat mengekstrak hampir semua informasi dari sebuah baris data.

Anggaplah kita memiliki baris dokumen HTML berikut ini.

```
>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 tutup >,
- setiap string yang tidak mengandung tanda kurung dengan
```

sub-pencocokan “(...)”,

```
 - kurung pembuka dan kurung penutup menggunakan solusi terpendek,
 - sekali lagi, semua string yang tidak mengandung tanda kurung,
 - dan sebuah kurung pembuka <.
```

Ekspresi reguler agak sulit untuk dipelajari tetapi sangat kuat.

```
>{pos,s,vt}=strxfind(line,">([^<>]+)<.+?>([^<>]+)<");
```

Hasilnya adalah posisi kecocokan, string yang cocok, dan vektor string untuk sub-cocokan.

```
>for k=1:length(vt); vt[k](), end;
```

```
1145.5
5.6
```

Berikut ini adalah 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
```

Situs web atau file dengan URL dapat dibuka di EMT dan dapat dibaca baris demi baris.

Dalam contoh, kita membaca versi saat ini dari situs EMT. Kami menggunakan ekspresi reguler untuk memindai "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 2024-01-12
```

Anda dapat menulis variabel dalam bentuk definisi Euler ke file atau ke baris perintah.

```
>writevar(pi,"mypi");
```

```
mypi = 3.141592653589793;
```

Untuk pengujian, kami membuat file Euler di direktori kerja EMT.

```
>file="test.e"; ...
>writevar(random(2,2),"M",file); ...
>printfile(file,3)
```

```
M = [ ..
0.5991820585590205, 0.7960280262224293;
0.5167243983231363, 0.2996684599070898];
```

Sekarang kita dapat memuat file tersebut. Ini akan mendefinisikan matriks M.

```
>load(file); show M,
```

```
M =
  0.59918   0.79603
  0.51672   0.29967
```

Sebagai catatan, jika writevar() digunakan pada sebuah variabel, maka ia akan mencetak definisi variabel dengan nama variabel tersebut.

```
>writevar(M); writevar(inch$)
```

```
M = [ ..
0.5991820585590205, 0.7960280262224293;
0.5167243983231363, 0.2996684599070898];
inch$ = 0.0254;
```

Kita juga dapat membuka file baru atau menambahkan ke file yang sudah ada. Dalam contoh ini, kami menambahkan ke file yang telah dibuat sebelumnya..

```
>open(file,"a"); ...
>writevar(random(2,2),"M1"); ...
>writevar(random(3,1),"M2"); ...
>close();
>load(file); show M1; show M2;
```

```
M1 =
  0.30287   0.15372
   0.7504   0.75401
M2 =
  0.27213
 0.053211
  0.70249
```

Untuk menghapus file, gunakan fileremove().

```
>fileremove(file);
```

Sebuah vektor baris dalam sebuah file tidak membutuhkan koma, jika setiap angka berada dalam baris baru. Mari kita buat file seperti itu, dengan menulis setiap baris satu per satu dengan writeln().

```
>open(file,"w"); writeln("M = ["); ...
>for i=1 to 5; writeln(""+random()); end; ...
>writeln("];"); close(); ...
>printfile(file)
```

```
M = [
0.344851384551
0.0807510017715
0.876519562911
0.754157709472
0.688392638934
];
```

```
>load(file); M
```

```
[0.34485,  0.080751,  0.87652,  0.75416,  0.68839]
```

Latihan Soal

1. Analisis data statistika deskriptif

```
>mean([50,70,76,75,77,68,45,30,15,71,69,84,15,88,90])
```

```
61.533
```

```
>data=[485, 360, 490, 900, 721, 840];
>urutan=sort(data)
```

```
[360,  485,  490,  721,  840,  900]
```

```
>median([urutan])
```

```
605.5
```

```
>data=[5, 9, 6, 10, 7, 8, 9,4,6,7];
>urut=sort(data)
```

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

```
>xbar=mean(urut)
```

```
7.1
```

```
>dev= urut-xbar
```

```
[-3.1,  -2.1,  -1.1,  -1.1,  -0.1,  -0.1,  0.9,  1.9,  1.9,  2.9]
```

```
>varians=mean(dev^2)
```

```
3.29
```

```
>data=[75,66,90,98,80,70,75,55,50,90];
>urut=sort(data)
```

```
[50,  55,  66,  70,  75,  75,  80,  90,  90,  98]
```

```
>x=mean(urut)
```

```
74.9
```

```
>dev=urut-x
```

```
[-24.9,  -19.9,  -8.9,  -4.9,  0.1,  0.1,  5.1,  15.1,  15.1,  23.1]
```

```
>varians=mean(dev^2)
```

```
213.49
```

```
>simpanganbaku= sqrt(varians)
```

```
14.611
```

```
>x=[30,60,78,90,96,74,55,79,90]; max(x)- min(x)
```

```
66
```

```
>data=[7,3,8,5,9,4,8,3,10,2,7,6,8,7,2,8,9,7,9,15,8,9];
>urut=sort(data)
```

```
[2,  2,  3,  3,  4,  5,  6,  7,  7,  7,  7,  8,  8,  8,  8,  8,  9,  9,
9,  9,  10,  15]
```

```
>quartiles(urut)
```

```
[2,  5,  7.5,  9,  15]
```

2. Uji statistik
contoh 1:
diberikan lemparan dadu acak sebanyak 1800 kali, tentukan nilai chi-kuadratnya apabila nilai harapan dadunya memiliki distribusi seragam

```
>throws=1800; sides=6;
>randomDice = intrandom(throws, sides);
>freqDice = multofsorted(sort(randomDice), 1:sides);
>expectDice = dup(throws/sides, sides)';
>chitest(freqDice, expectDice)
```

```
0.84915
```

contoh 2:
Diberikan hasil dari lemparan koin sebagai berikut, dengan G adalah gambar dan A adalah angka

G, A, A, G, A, A, A, G, A, A

Apakah koin tersebut adil?

```
>coinRes = "G, A, A, G, A, A, A, G, A, A";
>function chisquarecustom(x, y)...
```

```
sumSquare = 0;
for i=1 to length(x) step i;
   sumSquareCurr = (x[i] - y[i])^2 / y[i];
   sumSquare = sumSquare + sumSquareCurr;
end;
return sumSquare;
endfunction
```

```
>function replaceTokenToArr(strToken, separateToken, arrToken, arrSubsToken)...
```

```
tokens = strtokens(strToken, separateToken);
transformedArr = [];
for i = 1 to length(tokens) step i
   token = tokens[i];
   indexToken = indexof(arrToken, token);
   subs = arrSubsToken[indexToken];
   transformedArr = [transformedArr, subs];
end;
return transformedArr;
endfunction
```

```
>coinDis = replaceTokenToArr(coinRes, ", ", ["G", "A"], [1, 2])
```

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

```
>function getFrequencies(arr)...
```

```
uniqueArr = unique(arr);
sortArr = sort(arr);
firstIndexes = [];
freqs = [];
for i = 1 to length(uniqueArr) step i
   currUnique = uniqueArr[i];
   currFirstIndex = indexof(sortArr, currUnique) - 1;
   firstIndexes = [firstIndexes,  currFirstIndex];
end;
firstIndexes = [firstIndexes, length(arr)];
for i = 1 to (length(firstIndexes) - 1) step i
   currIndex = firstIndexes[i];
   nextIndex = firstIndexes[i + 1];
   currFreq = nextIndex - currIndex;
   freqs = [freqs, currFreq];
end;
return freqs;
endfunction
```

```
>coinFreq = getFrequencies(coinDis)
```

```
[3,  7]
```

```
>coinExpected = dup(sum(coinFreq)/length(coinFreq), length(coinFreq))'
```

```
[5,  5]
```

```
>chitest(coinFreq, coinExpected)
```

```
0.2059
```

Soal

Penelitian dilakukan untuk mengetahui partisipasi warga dalam suatu kepala desa yang dilihat dari jenis kelamin. Hasil penelitian tersebut diperoleh data sebagai berikut

Jenis Kelamin | Ikut | Tidak
Pria | 15 | 5
Wanita | 11 | 6

Apakah ada pengaruh jenis kelamin terhadap keikutseraan dalam pemilihan kepala desa?

```
>absenceGenders = [15,5;11,6]
```

```
        15          5
        11          6
```

Buat tabel agar mudah melihatnya

```
>writetable(absenceGenders, wc = 15, labr=["Pria", "Wanita"], labc = ["Ikut", "Tidak"])
```

```
                          Ikut          Tidak
           Pria             15              5
         Wanita             11              6
```

Memeriksa tabel untuk uji chi-kuadratnya

```
>tabletest(absenceGenders)
```

```
0.49478
```

Karena nilai terlalu besar, maka belum bisa menyimpulkan apakah terdapat hubungan antara keikutsertaan dengan gender

Lalu, dibuat tabel nilai harapan untuk melihat nilai harapan pada tabel tersebut

```
>writetable(expectedtable(absenceGenders), dc=1, wc=15, labr=["Pria", "Wanita"], labc=["Ikut", "Tida
```

```
                         Ikut          Tidak
          Pria           14.1            5.9
        Wanita           11.9            5.1
```

Terakhir, adalah menghitung nilai koefisien kontingensi

```
>contingency(absenceGenders)
```

```
0.15774
```

Contoh

diberikan nilai dua kelas berbeda, yaitu
Kelas A: 20, 30, 40, 50, 25, 35, 45, 55, 65, 45
Kelas B: 45, 74, 88, 91, 56, 68, 91, 31, 65, 68

Apakah nilai tengah antar keduanya sama?

Penyelesaian

Rumusan Hipotesis
Parameter populasi yang akan diuji adalah nilai antar kelas

$$\begin{array}{rl} \text{Hipotesis nol} & H_0 : \mu_a = \mu_b \\ \text{Hipotesis alternatif} & H_1 : \mu_a \neq \mu_b \end{array}$$

```
>function replaceTokenToArr2(str, replacement)...
```

```
tokens = strtokens(str, replacement);
transformedArr = [];
for i = 1 to length(tokens) step i
   token = evaluate(tokens[i]);
   transformedArr = [transformedArr, token];
end;
return transformedArr;
endfunction
```

```
>classAstr = "20, 30, 40, 50, 25, 35, 45, 55, 65, 45";
>classBstr = "45, 74, 88, 91, 56, 68, 91, 31, 65, 68";
>classA = replaceTokenToArr2(classAstr, ", ")
```

```
[20,  30,  40,  50,  25,  35,  45,  55,  65,  45]
```

```
>classB = replaceTokenToArr2(classBstr, ", ")
```

```
[45,  74,  88,  91,  56,  68,  91,  31,  65,  68]
```

```
>mediantest(classA, classB)
```

```
0.011507
```

Kesimpulan dari return fungsi:
Karena nilai return fungsi > 0.05 maka hipotesis nol diterima, artinya bahwa nilai tengah dari kedua kelas sama.

```
>ranktest(classA, classB)
```

```
0.0022932
```

Untuk melakukan pengujian lebih baik, dapat menggunakan fungsi ranktest

dimana hasil dari atas menunjukkan bahwa nilai tengah kedua kelas sama Karena nilai koefisien kontingensi mendekati 0, maka dapat disimpulkan bahwa hubungan antara keikutsertaan dengan jenis kelamin tidak ada (sangat lemah). Karena nilai error dari chi-kuadratnya kecil, maka kita dapat menolak hipotesis bahwa koin tersebut adil (setimbang).

3. Menggambar Grafik Statistika

Contoh Soal
Diketahui data ukuran sepatu siswa di suatu sekolahan

```
>A=[40,42,45,45,41,38,39,40,42,44,35,36,38,39,40,41,39,38,37,36,40,43,42,41,39,37,38,39,44,39,41,43]
```

```
[40,  42,  45,  45,  41,  38,  39,  40,  42,  44,  35,  36,  38,  39,
40,  41,  39,  38,  37,  36,  40,  43,  42,  41,  39,  37,  38,  39,
44,  39,  41,  43]
```

```
>boxplot(A):
```

Penjelasan diagram kotak tersebut adalah. Dapat diketahui nilai minimum sebesar 35, Q1=38, Median (Q2) = 40, Q3=42. Dan nilai maksimumnya adalah 45. Sebaran data tersebut juga diketahui simetris, karena si kuartil 1 dan 2 jaraknya sama dengan kuartil 2 dan 3.

Contoh lain perbandingan 15 simulasi 500 nilai terdistribusi normal menggunakan box plot dan ditemukan pencilan sbb

```
>p=normal(15,500); boxplot(p):
```

Contoh soal :
Kita akan membuat diagram batang secara random

```
>columnsplot(cumsum(random(6)),style="/",color=red):
>columnsplot(cumsum(random(15)),style="-",color=black):
>columnsplot(cumsum(random(3)),style="|",color=orange):
```

Kita akan membuat diagram batang dari penjualan warung watashi

```
>day=["Senin","Selasa","Rabu","Kamis","Jumat","Sabtu","Minggu"];
>values=[30,40,20,10,50,60,70];
>columnsplot(values,lab=day,color=yellow);
>title("Data Penjualan Perhari di Toko Watashi")
>insimg
>CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.9,0,0)]
```

```
[5.8753e+07,  2,  15,  3,  6.5405e+07]
```

```
>i=[1,2,3,4,5]; piechart(values[i],color=CP[i],lab=day[i]):
>starplot(normal(1,15)+16,lab=1:15,>rays):
>starplot(values,lab=day,>rays):
>plot2d(makeimpulse(1:20,random(1,20)),>bar):
>logimpulseplot(1:20,-log(random(1,20))*10):
>aspect(1); plot2d(random(100),>histogram):
>r=150:5:185; v=[22,71,136,150,139,71,32];
>plot2d(r,v,a=150,b=185,c=0,d=150,bar=1,style="/"):
```

```
>plot2d("qnormal(x,0,1)",-5,5); ...
>plot2d("qnormal(x,0,1)",a=1,b=4,>add,>filled):
```

Contoh kurva fungsi distribusi kumulatif kontinu terdiri atas tiga bagian yaitu
1. Bernilai 0 untuk x dibawah minimal dari daerah rentang
2. Merupakan fungsi monoton naik pada daerah rentang
3. Mempunyai nilai konstan 1 diatas batas maksimum daerah rentangnya

```
>x=normal(1,6);
```

Baris tersebut akan menghasilkan suatu nilai acak dari distribusi normal dengan mean 1 dan standar deviasi 6, dan nilai tsb disimpan dalam var x.

```
>xs=sort(x);
>plot2d("empdist",-3,5;xs):
```

Grafik fungsi distribusi kumulatif peubah acak diskrit merupakan fungsi tangga dengan terendah 0 dan tertinggi 1.

```
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 of
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 for
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