KUMPULAN MATERI PENGGUNAAN EMT (Euler Math Toolbox)

Tugas Ini Disusun Untuk Memenuhi Tugas Mata Kuliah Aplikasi Komputer

Dosen Pengampu:
Bapak Drs. Sahid M.Sc dan Thesa Adi Saputra Yusri M.Cs

Disusun oleh:
Diva Nagita
23030630024
Matematika B 2024

PRODI MATEMATIKA
DEPARTEMEN PENDIDIKAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI YOGYAKARTA
2024

DAFTAR ISI

unicodehyperref hyphensurl [ ]book xcolor amsmath,amssymb iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype [protrusion]basicmath parskip bookmark xurl same hidelinks, pdfcreator=LaTeX via pandoc

PENDAHULUAN DAN PENGENALAN CARA KERJA EMT

Selamat datang! Ini adalah pengantar pertama ke Euler Math Toolbox (disingkat EMT atau Euler). EMT adalah sistem terintegrasi yang merupakan perpaduan kernel numerik Euler dan program komputer aljabar Maxima.

- Bagian numerik, GUI, dan komunikasi dengan Maxima telah dikembangkan oleh R. Grothmann, seorang profesor matematika di Universitas Eichstätt, Jerman. Banyak algoritma numerik dan pustaka software opensource yang digunakan di dalamnya.
- Maxima adalah program open source yang matang dan sangat kaya untuk perhitungan simbolik dan aritmatika tak terbatas. Software ini dikelola oleh sekelompok pengembang di internet.
- Beberapa program lain (LaTeX, Povray, Tiny C Compiler, Python) dapat digunakan di Euler untuk memungkinkan perhitungan yang lebih cepat maupun tampilan atau grafik yang lebih baik.

Yang sedang Anda baca (jika dibaca di EMT) ini adalah berkas notebook di EMT. Notebook aslinya bawaan EMT (dalam bahasa Inggris) dapat dibuka melalui menu File, kemudian pilih "Open Tutorias and Example", lalu pilih file "00 First Steps.en". Perhatikan, file notebook EMT memiliki ekstensi ".en". Melalui notebook ini Anda akan belajar menggunakan software Euler untuk menyelesaikan berbagai masalah matematika.

Panduan ini ditulis dengan Euler dalam bentuk notebook Euler, yang berisi teks (deskriptif), baris-baris perintah, tampilan hasil perintah (numerik, ekspresi matematika, atau gambar/plot), dan gambar yang disisipkan dari file gambar.

Untuk menambah jendela EMT, Anda dapat menekan [F11]. EMT akan menampilkan jendela grafik di layar desktop Anda. Tekan [F11] lagi untuk kembali ke tata letak favorit Anda. Tata letak disimpan untuk sesi berikutnya.

Anda juga dapat menggunakan [Ctrl]+[G] untuk menyembunyikan jendela grafik. Selanjutnya Anda dapat beralih antara grafik dan teks dengan tombol [TAB].

Seperti yang Anda baca, notebook ini berisi tulisan (teks) berwarna hijau, yang dapat Anda edit dengan mengklik kanan teks atau tekan menu Edit -> Edit Comment atau tekan [F5], dan juga baris perintah EMT yang ditandai dengan ">" dan berwarna merah. Anda dapat menyisipkan baris perintah baru dengan cara menekan tiga tombol bersamaan: [Shift]+[Ctrl]+[Enter].

Komentar (Teks Uraian)

Komentar atau teks penjelasan dapat berisi beberapa "markup" dengan sintaks sebagai berikut.

- – Judul
- ** Sub-Judul
- latex: F (x) = ∫ _a $x f(t), dtmathjax : \frac{x^2-1}{x-1} = x + 1$
- maxima: 'integrate(x^3,x) = integrate(x^3,x) + C

- http://www.euler-math-toolbox.de

- See: http://www.google.de | Google

- image: hati.png
  \- —

Hasil sintaks-sintaks di atas (tanpa diawali tanda strip) adalah sebagai berikut.

Judul

Sub-Judul

$$F(x) = \int_a^x f(t)\,dt$$

$$\frac{x^2-1}{x-1} = x+1$$

$$\int x^3\,dx = C + \frac{x^4}{4}$$

http://www.euler-math-toolbox.de
Google
image: hati.png
>// baris perintah diawali dengan >, komentar (keterangan) diawali dengan //

Baris Perintah

Mari kita tunjukkan cara menggunakan EMT sebagai kalkulator yang sangat canggih.

EMT berorientasi pada baris perintah. Anda dapat menuliskan satu atau lebih perintah dalam satu baris perintah. Setiap perintah harus diakhiri dengan koma atau titik koma.

- Titik koma menyembunyikan output (hasil) dari perintah.
- Sebuah koma mencetak hasilnya.
- Setelah perintah terakhir, koma diasumsikan secara otomatis (boleh tidak ditulis).

Dalam contoh berikut, kita mendefinisikan variabel r yang diberi nilai 1,25. Output dari definisi ini adalah nilai variabel. Tetapi karena tanda titik koma, nilai ini tidak ditampilkan. Pada kedua perintah di belakangnya, hasil kedua perhitungan tersebut ditampilkan.
>r=1.25; pi*r^2, 2*pi*r

```
4.90873852123
7.85398163397
```

Latihan

- Sisipkan beberapa baris perintah baru
- Tulis perintah-perintah baru untuk melakukan suatu perhitungan yang Anda inginkan, boleh menggunakan variabel, boleh tanpa variabel.

>(785-3)^2-1000

```
610524
```

>200-3+77

```
274
```

>55+6*3

```
73
```

>O=12749; 48^2-O

```
-10445
```

Beberapa catatan yang harus Anda perhatikan tentang penulisan sintaks perintah EMT.

- Pastikan untuk menggunakan titik desimal, bukan koma desimal untuk bilangan!
- Gunakan * untuk perkalian dan ^ untuk eksponen (pangkat).
- Seperti biasa, * dan / bersifat lebih kuat daripada + atau -.
- ^ mengikat lebih kuat dari *, sehingga pi* r ^ 2 merupakan rumus luas lingkaran.
- Jika perlu, Anda harus menambahkan tanda kurung, seperti pada 2 ^ (2 ^ 3).

Perintah r = 1.25 adalah menyimpan nilai ke variabel di EMT. Anda juga dapat menulis r: = 1.25 jika mau. Anda dapat menggunakan spasi sesuka Anda.

Anda juga dapat mengakhiri baris perintah dengan komentar yang diawali dengan dua garis miring (//).

>r := 1.25 // Komentar: Menggunakan := sebagai ganti =

```
1.25
```

Argumen atau input untuk fungsi ditulis di dalam tanda kurung.

>sin(45°), cos(pi), log(sqrt(E))

```
0.707106781187
-1
0.5
```

Seperti yang Anda lihat, fungsi trigonometri bekerja dengan radian, dan derajat dapat diubah dengan °. Jika keyboard Anda tidak memiliki karakter derajat tekan [F7], atau gunakan fungsi deg() untuk mengonversi.

EMT menyediakan banyak sekali fungsi dan operator matematika.Hampir semua fungsi matematika sudah tersedia di EMT. Anda dapat melihat daftar lengkap fungsi-fungsi matematika di EMT pada berkas Referensi (klik menu Help -> Reference)

Untuk membuat rangkaian komputasi lebih mudah, Anda dapat merujuk ke hasil sebelumnya dengan "%". Cara ini sebaiknya hanya digunakan untuk merujuk hasil perhitungan dalam baris perintah yang sama.

>(sqrt(5)+1)/2, %^2-%+1 // Memeriksa solusi x^2-x+1=0

```
1.61803398875
2
```

Latihan

- Buka berkas Reference dan baca fungsi-fungsi matematika yang tersedia di EMT.
- Sisipkan beberapa baris perintah baru.
- Lakukan contoh-contoh perhitungan menggunakan fungsi-fungsi matematika di EMT.

>sin(35°)

```
0.573576436351
```

>log10(400)

```
2.60205999133
```

>cot(45)

```
0.617369623784
```

>mod(45,3)

```
0
```

>ceil(0.012234465657)

```
1
```

Satuan

EMT dapat mengubah unit satuan menjadi sistem standar internasional (SI). Tambahkan satuan di belakang angka untuk konversi sederhana.

>1miles // 1 mil = 1609,344 m

```
1609.344
```

Beberapa satuan yang sudah dikenal di dalam EMT adalah sebagai berikut. Semua unit diakhiri dengan tanda dolar ($), namun boleh tidak perlu ditulis dengan mengaktifkan easyunits.

kilometer$:=1000;

km:= $kilometer$;

cm$:=0.01;

mm$:=0.001;

minute$:=60;

min:= $minute$;

minutes:= $minute$;

hour:= $60 * minute$;

h:= $hour$;

hours:= $hour$;

day:= $24 * hour$;

days:= day;

d:= day;

year:= $365.2425 * day$;

years:= $year$;

y:= $year$;

inch$:=0.0254;

in:= $inch$;

feet:= $12 * inch$;

foot:= $feet$;

ft:= $feet$;
yard:= $3 * feet$;
yards:= $yard$;
yd:= $yard$;
mile:= $1760 * yard$;
miles:= $mile$;
kg$:=1;
sec$:=1;
ha$:=10000;
Ar$:=100;
Tagwerk$:=3408;
Acre$:=4046.8564224;
pt$:=0.376mm;
Untuk konversi ke dan antar unit, EMT menggunakan operator khusus, yakni ->.
>4km -> miles, 4inch -> " mm"

```
2.48548476895
101.6 mm
```

Format Tampilan Nilai

Akurasi internal untuk nilai bilangan di EMT adalah standar IEEE, sekitar 16 digit desimal. Aslinya, EMT tidak mencetak semua digit suatu bilangan. Ini untuk menghemat tempat dan agar terlihat lebih baik. Untuk mengatrtamilan satu bilangan, operator berikut dapat digunakan.
>pi

```
3.14159265359
```

>longest pi

```
      3.141592653589793
```

>long pi

```
3.14159265359
```

>short pi

```
3.1416
```

>shortest pi

```
   3.1
```

>fraction pi

```
312689/99532
```

>short 1200*1.03^10, long E, longest pi

```
1612.7
2.71828182846
      3.141592653589793
```

Format aslinya untuk menampilkan nilai menggunakan sekitar 10 digit. Format tampilan nilai dapat diatur secara global atau hanya untuk satu nilai.

Anda dapat mengganti format tampilan bilangan untuk semua perintah selanjutnya. Untuk mengembalikan ke format aslinya dapat digunakan perintah "defformat" atau "reset".

>longestformat; pi, defformat; pi

```
3.141592653589793
3.14159265359
```

Kernel numerik EMT bekerja dengan bilangan titik mengambang (floating point) dalam presisi ganda IEEE (berbeda dengan bagian simbolik EMT). Hasil numerik dapat ditampilkan dalam bentuk pecahan.

>1/7+1/4, fraction %

```
0.392857142857
11/28
```

Perintah Multibaris

Perintah multi-baris membentang di beberapa baris yang terhubung dengan ". . . " di setiap akhir baris, kecuali baris terakhir. Untuk menghasilkan tanda pindah baris tersebut, gunakan tombol [Ctrl]+[Enter]. Ini akan menyambung perintah ke baris berikutnya dan menambahkan ". . . " di akhir baris sebelumnya. Untuk menggabungkan suatu baris ke baris sebelumnya, gunakan [Ctrl]+[Backspace].

Contoh perintah multi-baris berikut dapat dijalankan setiap kali kursor berada di salah satu barisnya. Ini juga menunjukkan bahwa . . . harus berada di akhir suatu baris meskipun baris tersebut memuat komentar.

>a=4; b=15; c=2; // menyelesaikan a*xˆ2+b*x+c=0 secara manual . . .
> D=sqrt(b$^{2/(a}$2*4)-c/a); . . .
> -b/(2*a) + D, . . .
> -b/(2*a) - D

```
-0.138444501319
-3.61155549868
```

Menampilkan Daftar Variabe

Untuk menampilkan semua variabel yang sudah pernah Anda definisikan sebelumnya (dan dapat dilihat kembali nilainya), gunakan perintah "listvar".

>listvar

```
r                   1.25
O                   12749
a                   4
b                   15
c                   2
D                   1.73655549868123
```

Perintah listvar hanya menampilkan variabel buatan pengguna. Dimungkinkan untuk menampilkan variabel lain, dengan menambahkan string termuat di dalam nama variabel yang diinginkan.

Perlu Anda perhatikan, bahwa EMT membedakan huruf besar dan huruf kecil. Jadi variabel "d" berbeda dengan variabel "D".

Contoh berikut ini menampilkan semua unit yang diakhiri dengan "m" dengan mencari semua variabel yang berisi "m$".

>listvar m$

```
km$                 1000
cm$                 0.01
mm$                 0.001
nm$                 1853.24496
gram$               0.001
m$                  1
hquantum$           6.62606957e-34
atm$                101325
```

Untuk menghapus variabel tanpa harus memulai ulang EMT gunakan perintah "remvalue".

>remvalue a,b,c,D

>D

```
Variable D not found!
Error in:
D ...
 ^
```

Menampilkan Panduan

Untuk mendapatkan panduan tentang penggunaan perintah atau fungsi di EMT, buka jendela panduan dengan menekan [F1] dan cari fungsinya. Anda juga dapat mengklik dua kali pada fungsi yang tertulis di baris perintah atau di teks untuk membuka jendela panduan.

Coba klik dua kali pada perintah "intrandom" berikut ini!

>intrandom(10,6)

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

Di jendela panduan, Anda dapat mengklik kata apa saja untuk menemukan referensi atau fungsi.

Misalnya, coba klik kata "random" di jendela panduan. Kata tersebut boleh ada dalam teks atau di bagian "See:" pada panduan. Anda akan menemukan penjelasan fungsi "random", untuk menghasilkan bilangan acak berdistribusi uniform antara 0,0 dan 1,0. Dari panduan untuk "random" Anda dapat menampilkan panduan untuk fungsi "normal", dll.

>random(10)

```
[0.270906,  0.704419,  0.217693,  0.445363,  0.308411,  0.914541,
0.193585,  0.463387,  0.095153,  0.595017]
```

>normal(10)

```
[-0.495418,  1.6463,  -0.390056,  -1.98151,  3.44132,  0.308178,
-0.733427,  -0.526167,  1.10018,  0.108453]
```

Matriks dan Vektor

EMT merupakan suatu aplikasi matematika yang mengerti "bahasa matriks". Artinya, EMT menggunakan vektor dan matriks untuk perhitungan-perhitungan tingkat lanjut. Suatu vektor atau matriks dapat didefinisikan dengan tanda kurung siku. Elemen-elemennya dituliskan di dalam tanda kurung siku, antar elemen dalam satu baris dipisahkan oleh koma(,), antar baris dipisahkan oleh titik koma (;).

Vektor dan matriks dapat diberi nama seperti variabel biasa.

>v=[4,5,6,3,2,1]

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

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

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

Karena EMT mengerti bahasa matriks, EMT memiliki kemampuan yang sangat canggih untuk melakukan perhitungan matematis untuk masalah-masalah aljabar linier, statistika, dan optimisasi.

Vektor juga dapat didefinisikan dengan menggunakan rentang nilai dengan interval tertentu menggunakan tanda titik dua (:),seperti contoh berikut ini.

>c=1:5

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

>w=0:0.1:1

```
[0,  0.1,  0.2,  0.3,  0.4,  0.5,  0.6,  0.7,  0.8,  0.9,  1]
```

>mean(w^2)

```
0.35
```

Bilangan Kompleks

EMT juga dapat menggunakan bilangan kompleks. Tersedia banyak fungsi untuk bilangan kompleks di EMT. Bilangan imaginer

dituliskan dengan huruf I (huruf besar I), namun akan ditampilkan dengan huruf i (i kecil).

re(x) : bagian riil pada bilangan kompleks x.
im(x) : bagian imaginer pada bilangan kompleks x.
complex(x) : mengubah bilangan riil x menjadi bilangan kompleks.
conj(x) : Konjugat untuk bilangan bilangan komplkes x.
arg(x) : argumen (sudut dalam radian) bilangan kompleks x.
real(x) : mengubah x menjadi bilangan riil.

Apabila bagian imaginer x terlalu besar, hasilnya akan menampilkan pesan kesalahan.

>sqrt(-1) // Error!
>sqrt(complex(-1))

>z=2+3*I, re(z), im(z), conj(z), arg(z), deg(arg(z)), deg(arctan(3/2))

```
2+3i
2
3
2-3i
0.982793723247
56.309932474
56.309932474
```

>deg(arg(I)) // 90°

```
90
```

>sqrt(-1)

```
Floating point error!
Error in sqrt
Error in:
sqrt(-1) ...
        ^
```

>sqrt(complex(-1))

```
0+1i
```

EMT selalu menganggap semua hasil perhitungan berupa bilangan riil dan tidak akan secara otomatis mengubah ke bilangan kompleks.

Jadi akar kuadrat -1 akan menghasilkan kesalahan, tetapi akar kuadrat kompleks didefinisikan untuk bidang koordinat dengan cara seperti biasa. Untuk mengubah bilangan riil menjadi kompleks, Anda dapat menambahkan 0i atau menggunakan fungsi "complex".

>complex(-1), sqrt(%)

```
-1+0i
0+1i
```

Matematika Simbolik

EMT dapat melakukan perhitungan matematika simbolis (eksak) dengan bantuan software Maxima. Software Maxima otomatis sudah terpasang di komputer Anda ketika Anda memasang EMT. Meskipun demikian, Anda dapat juga memasang software Maxima tersendiri (yang terpisah dengan instalasi Maxima di EMT).

Pengguna Maxima yang sudah mahir harus memperhatikan bahwa terdapat sedikit perbedaan dalam sintaks antara sintaks asli Maxima dan sintaks ekspresi simbolik di EMT.

Untuk melakukan perhitungan matematika simbolis di EMT, awali perintah Maxima dengan tanda "&". Setiap ekspresi yang dimulai dengan "&" adalah ekspresi simbolis dan dikerjakan oleh Maxima.

>&(a+b)^2

```
                                        2
                                 (b + a)
```

>&expand((a+b)^2), &factor(x^2+5*x+6)

```
                              2             2
                             b  + 2 a b + a


                             (x + 2) (x + 3)
```

>&solve(a*x^2+b*x+c,x) // rumus abc

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

>&(a^2-b^2)/(a+b), &ratsimp(%) // ratsimp menyederhanakan bentuk pecahan

```
                                  2    2
                                 a  - b
                                 -------
                                  b + a


                                  a - b
```

>10! // nilai faktorial (modus EMT)

```
3628800
```

>&10! //nilai faktorial (simbolik dengan Maxima)

```
                                   3628800
```

Untuk menggunakan perintah Maxima secara langsung (seperti perintah pada layar Maxima) awali perintahnya dengan tanda "::" pada baris perintah EMT. Sintaks Maxima disesuaikan dengan sintaks EMT (disebut "modus kompatibilitas").

>factor(1000) // mencari semua faktor 1000 (EMT)

```
[2,  2,  2,  5,  5,  5]
```

>:: factor(1000) // faktorisasi prima 1000 (dengan Maxima)

```
                                     3  3
                                    2  5
```

>:: factor(20!)

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

Jika Anda sudah mahir menggunakan Maxima, Anda dapat menggunakan sintaks asli perintah Maxima dengan menggunakan tanda ":::" untuk mengawali setiap perintah Maxima di EMT. Perhatikan, harus ada spasi antara ":::" dan perintahnya.

>::: binomial(5,2); // nilai C(5,2)

```
                                   10
```

>::: binomial(m,4); // C(m,4)=m!/(4!(m-4)!)

```
                        (m - 3) (m - 2) (m - 1) m
                        -------------------------
                                   24
```

>::: trigexpand(cos(x+y)); // rumus cos(x+y)=cos(x) cos(y)-sin(x)sin(y)

```
                      cos(x) cos(y) - sin(x) sin(y)
```

>::: trigsimp(((1-sin(x)$^{2)*cos(x))/cos(x)}$2+tan(x)*sec(x)ˆ2) //menyederhanakan fungsi trigonometri

```
                                     4
                           sin(x) + cos (x)
                           ----------------
                                  3
                               cos (x)
```

Untuk menyimpan ekspresi simbolik ke dalam suatu variabel digunakan tanda "&=".

>p1 &= (xˆ3+1)/(x+1)

```
 3
x  + 1
------
x + 1
```

>&ratsimp(p1)

```
 2
x  - x + 1
```

Untuk mensubstitusikan suatu nilai ke dalam variabel dapat digunakan perintah “with”.
>&p1 with x=3 // (3^3+1)/(3+1)

```
7
```

>&p1 with x=a+b, &ratsimp(%) //substitusi dengan variabel baru

```
       3
(b + a)  + 1
------------
 b + a + 1
```

```
 2                  2
b  + (2 a - 1) b + a  - a + 1
```

>&diff(p1,x) //turunan p1 terhadap x

```
   2       3
3 x       x  + 1
----- - --------
x + 1          2
        (x + 1)
```

>&integrate(p1,x) // integral p1 terhadap x

```
   3      2
2 x  - 3 x  + 6 x
-----------------
        6
```

Tampilan Matematika Simbolik dengan LaTeX

Anda dapat menampilkan hasil perhitunagn simbolik secara lebih bagus menggunakan LaTeX. Untuk melakukan hal ini, tambahkan tanda dolar ($) di depan tanda & pada setiap perintah Maxima.

Perhatikan, hal ini hanya dapat menghasilkan tampilan yang diinginkan apabila komputer Anda sudah terpasang software LaTeX.
>$&(a+b)^2

$$(b+a)^2$$

>$&expand((a+b)^2), $&factor(x^2+5*x+6)

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

$$(x+2)\ (x+3)$$

>$&solve(a*x^2+b*x+c,x) // rumus abc

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

>$&(a^2-b^2)/(a+b), $&ratsimp(%)

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

$$a-b$$

Selamat Belajar dan Berlatih!

Baik, itulah sekilas pengantar penggunaan software EMT. Masih banyak kemampuan EMT yang akan Anda pelajari dan praktikkan.

Sebagai latihan untuk memperlancar penggunaan perintah-perintah EMT yang sudah dijelaskan di atas, silakan Anda lakukan hal-hal sebagai berikut. - Carilah soal-soal matematika dari buku-buku Matematika. - Tambahkan beberapa baris perintah EMT pada notebook ini. - Selesaikan soal-soal matematika tersebut dengan menggunakan EMT. - Pilih soal-soal yang sesuai dengan perintah-perintah yang sudah dijelaskan dan dicontohkan di atas. —

1.tentukan turunan dari persamaan berikut

>G &= (5*x^7-2)*(6*x^2+2)

```
                    2           7
            (6 x  + 2) (5 x  - 2)
```

>&diff(G,x) //turunan G terhadap x

```
                 7                 6        2
      12 x (5 x  - 2) + 35 x  (6 x  + 2)
```

2. Carilah integral dari persamaan berikut dengan batas bawah x=0 dan batas atas x=3

>K &= 4*x^3

```
                                 3
                              4 x
```

>&integrate(K,x,0,3)

```
                               81
```

3. Carilah integral persamaan berikut terhadap x dengan batas bawah x=1 sampai x=pi

>&integrate(3*cos(x),x,1,pi)

```
                         - 3 sin(1)
```

4. Tentukan turunan ketiga dari fumgsi berikut

>N &= x^3-3*x^2+8

```
                          3       2
                         x  - 3 x  + 8
```

>M &= diff(N,x) //turunan pertama dari N

$$3x^2 - 6x$$

>T &= diff(M,x) //turunan kedua dari N

$$6x - 6$$

>&diff(T,x) //turunan ketiga dari N

$$6$$

upquote []microtype [protrusion]basicmath parskip bookmark xurl same hidelinks, pdfcreator=LaTeX via pandoc

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

Menyederhanakan bentuk aljabar:

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

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

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

Menjabarkan:

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

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

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

Baris Perintah

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

Baris perintah berikut ini hanya akan mencetak hasil dari ekspresi, bukan penugasan atau perintah format.

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

```
16.7551608191
```

Perintah harus dipisahkan dengan tanda kosong. Baris perintah berikut ini 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 return. Jadi, Anda 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 tersebut 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 kedua baris.

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

>%+ x=4+5; ...
> //Garis ini tidak akan terlihat setelah kursor keluar dari garis

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

```
81
```

Euler mendukung perulangan dalam baris perintah, selama perulangan tersebut masuk ke dalam satu baris tunggal atau beberapa baris. Dalam program, tentu saja pembatasan ini tidak berlaku. Untuk informasi lebih lanjut, baca pengantar berikut ini.

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

```
1.5
1.41666666667
1.41421568627
1.41421356237
1.41421356237
```

Tidak masalah untuk menggunakan multi-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 bersyarat juga bisa digunakan.

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

```
Thought so!
```

Ketika Anda menjalankan perintah, kursor dapat berada di posisi mana pun dalam 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.

Ketika Anda menggerakkan kursor di sepanjang baris, pasangan tanda kurung atau tanda kurung pembuka dan penutup akan disorot. Juga, perhatikan baris status. Setelah tanda kurung pembuka dari 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 yang akan dicari. Pada baris kosong, bantuan untuk jendela bantuan akan ditampilkan.

Anda dapat menekan escape untuk mengosongkan baris, atau 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 mouse atau gunakan shift bersamaan dengan tombol kursor. Selain itu, Anda dapat menyalin tanda kurung yang disorot.

Sintaksis Dasar

Euler mengetahui fungsi matematika yang biasa. 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 dalam Euler. Tentu saja, x^(1/2) juga dapat digunakan.

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

Beberapa perintah dalam satu baris dipisahkan dengan "," atau ";". Titik koma menekan output dari perintah. Pada 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 mengatur 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 yang rumit seperti

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

Anda harus memasukkannya dalam bentuk 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. EMT membantu Anda dengan menyorot ekspresi yang diselesaikan oleh tanda kurung penutup. Anda juga harus memasukkan nama "pi" untuk huruf Yunani pi.

Hasil dari perhitungan ini adalah angka floating point. Secara default dicetak dengan akurasi sekitar 12 digit. Pada baris perintah berikut, kita juga mempelajari bagaimana kita dapat 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 terbuat dari operator dan fungsi. Jika perlu, ekspresi tersebut harus mengandung tanda kurung untuk memaksa urutan eksekusi yang benar. Jika ragu, mengatur tanda kurung adalah ide yang bagus. Perhatikan bahwa EMT menampilkan tanda kurung pembuka dan penutup saat mengedit baris perintah.

>(cos(pi/4)+1)$^{3*(\sin(pi/4)+1)}$2

```
14.4978445072
```

Operator numerik Euler meliputi

- unary atau operator plus
- unary atau operator minus
  *, /
  . produk matriks
  pangkat a^b untuk a positif atau bilangan bulat b (a**b juga bisa

digunakan)

n! operator faktorial

dan masih banyak lagi.

Berikut adalah 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

Beberapa perintah memiliki alias, misalnya ln untuk log.

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

```
2
0.5
```

Contoh

>sin(30°)

```
0.5
```

Pastikan untuk menggunakan tanda kurung (tanda kurung bulat), apabila ada keraguan tentang urutan eksekusi! Berikut ini tidak sama dengan (2$^{3)}$4, yang merupakan default untuk 2^{3}4 di EMT (beberapa sistem numerik melakukannya dengan cara lain).
>cos(sin(0°))

```
1
```

>sin(pi/9)

```
0.342020143326
```

>cos(7*pi/18)

```
0.342020143326
```

>sin(42°)

```
0.669130606359
```

>2^{6}2, (2$^{3)}$4, 2$^{(3}$4)

```
68719476736
4096
2.41785163923e+24
```

>cos(45°)

```
0.707106781187
```

>cos(495°)

```
-0.707106781187
```

>sec(-270°)

```
-5.44374645107e+15
```

>tan(-600°)

```
-1.73205080757
```

>sin(150°)

```
0.5
```

Menyederhanakan

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

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

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

>2^{3}4, (2$^{3)}$4, 2$^{(3}$4)

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

Bilangan Real

Tipe data utama dalam Euler adalah bilangan real. Bilangan real direpresentasikan dalam format IEEE dengan akurasi sekitar 16 digit desimal.
>longest 1/3

```
     0.3333333333333333
```

Representasi ganda internal membutuhkan 8 byte.
>printdual(1/3)

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

Contoh

>longest 22/7

```
      3.142857142857143
```

>printdual(22/7)

```
1.1001001001001001001001001001001001001001001001001001*2^1
```

>printhex(22/7)

```
3.2492492492492*16^0
```

>printhex(1/3)

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

>longest 1/3

```
     0.3333333333333333
```

String

String dalam Euler didefinisikan dengan "...".
>"A string can contain anything."

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

String dapat digabungkan dengan | atau dengan +. Ini juga berfungsi dengan angka, yang dikonversi menjadi string dalam kasus tersebut.
>"The area of the circle with radius" + 2 + " cm is " + pi*4 + " cm^2."

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

Fungsi cetak juga mengonversi angka ke string. Fungsi ini dapat mengambil sejumlah digit dan sejumlah tempat (0 untuk output padat), dan secara optimal satu unit.
>"Golden Ratio :" + print((1+sqrt(5))/2,5,0)

```
Golden Ratio : 1.61803
```

Ada string khusus tidak ada, yang tidak mencetak. Dikembalikan oleh beberapa fungsi, ketika hasilnya tidak penting. (Dikembalikan secara otomatis, jika fungsi tidak memiliki pernyataan pengembalian).

>none

Untuk mengonversi 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 digabungkan.

>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 digabungkan seperti string lainnya.

>u" = " + 45 + u"°" // pdfLaTeX mungkin gagal menampilkan secara benar

```
 = 45°
```

I

Dalam komentar, entitas yang sama seperti , dll. dapat digunakan. Ini bisa menjadi alternatif yang cepat untuk Latex. (Detail lebih lanjut tentang komentar di bawah).

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

>v=strtochar(u"Ä 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 sebuah vektor angka Unicode. Fungsi kebalikannya adalah chartoutf().

>v[1]=strtochar(u"Ü")[1]; chartoutf(v)

```
Ü is a German letter
```

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

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

```
We have =.
```

Dimungkinkan juga untuk menggunakan entitas numerik.

>u"Ähnliches"

```
Ähnliches
```

Contoh

>“sin(5*pi/6)adalah” + print(sin(5*pi/6))

```
sin(5*pi/6)adalah      0.50
```

>“sin(24°)adalah” + print(sin(24°))

```
sin(24°)adalah      0.41
```

>“cos(3*pi)adalah” + print(cos(3*pi))

```
cos(3*pi)adalah     -1.00
```

>“Universitas Negeri Yogyakarta”

```
Universitas Negeri Yogyakarta
```

>“Aplikasi Komputer”

```
Aplikasi Komputer
```

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. Pada contoh, kita menggunakan kondisional isprime(n).

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

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

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

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

Contoh

>(1:30)>25, nonzeros(%)

```
[0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
0,  0,  0,  0,  0,  0,  0,  1,  1,  1,  1,  1]
[26,  27,  28,  29,  30]
```

>2<2

```
0
```

> “matematika”<“statistika”

```
1
```

>D=2|1:3:100

```
[2,  1,  4,  7,  10,  13,  16,  19,  22,  25,  28,  31,  34,  37,  40,
43,  46,  49,  52,  55,  58,  61,  64,  67,  70,  73,  76,  79,  82,
85,  88,  91,  94,  97,  100]
```

>E=5|1:10:1000

```
[5,  1,  11,  21,  31,  41,  51,  61,  71,  81,  91,  101,  111,  121,
131,  141,  151,  161,  171,  181,  191,  201,  211,  221,  231,  241,
251,  261,  271,  281,  291,  301,  311,  321,  331,  341,  351,  361,
371,  381,  391,  401,  411,  421,  431,  441,  451,  461,  471,  481,
491,  501,  511,  521,  531,  541,  551,  561,  571,  581,  591,  601,
611,  621,  631,  641,  651,  661,  671,  681,  691,  701,  711,  721,
731,  741,  751,  761,  771,  781,  791,  801,  811,  821,  831,  841,
851,  861,  871,  881,  891,  901,  911,  921,  931,  941,  951,  961,
971,  981,  991]
```

>D[nonzeros(isprime(D))]

```
[2,  7,  13,  19,  31,  37,  43,  61,  67,  73,  79,  97]
```

Format Keluaran

Format output default EMT mencetak 12 digit. Untuk memastikan bahwa kita melihat format default, kita atur ulang formatnya.
>defformat; pi

```
3.14159265359
```

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

```
      3.141592653589793
```

Berikut ini adalah representasi heksadesimal internal dari angka ganda.
>printhex(pi)

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

Format output dapat diubah secara permanen dengan perintah format.

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

```
    0.33333
    3.14159
    0.84147
```

Standarnya adalah format(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). Tetapi ini dapat diubah.

>setscalarformat(5); pi

```
3.1416
```

Fungsi "longestformat" juga menetapkan format skalar.

>longestformat; pi

```
3.141592653589793
```

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

format terpendek format terpendek format panjang, format terpanjang
format (panjang, digit) format bagus (panjang)
format pecahan (panjang)
defformat

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

Tetapi format keluaran 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 singkat untuk mencetak hasil dalam format pecahan. Kami sudah menggunakannya di atas.

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

```
25/12
```

Karena format internal menggunakan cara biner untuk menyimpan angka, maka nilai 0,1 tidak akan terwakili dengan tepat. Kesalahan bertambah sedikit, seperti yang Anda lihat dalam perhitungan berikut ini.

>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
```

Tetapi, dengan "longformat" default, Anda tidak akan melihat hal ini. Untuk 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
```

Contoh

>fraction 5+1/2+2/3+1/3

```
13/2
```

>format(10,8); pi

```
3.1415927
```

>format(12,8); sin(4)

```
-0.75680250
```

>longest pi^2/3

```
      3.289868133696453
```

>format(12,8); 1/9, pi

```
 0.11111111
 3.14159265
```

Ekspresi

String atau nama dapat digunakan untuk menyimpan ekspresi matematika, yang dapat dievaluasi oleh EMT. Untuk ini, gunakan tanda kurung setelah ekspresi. Jika Anda bermaksud menggunakan string sebagai ekspresi, gunakan konvensi untuk menamainya "fx" atau "fxy", dll. Ekspresi 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 dengan menggunakan parameter yang ditetapkan.

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

```
-0.91953576
```

Perhatikan bahwa ekspresi akan selalu menggunakan variabel global, meskipun ada variabel dalam 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*xˆ2",3,5) // computes 4*3ˆ2 not 5*3ˆ2

```
36.00000000
```

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*xˆ2",3,5)

```
45.00000000
```

Sebagai referensi, kami menyatakan 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*xˆ2",at=5}},3)

```
45.00000000
```

Ekspresi dalam x sering digunakan seperti halnya fungsi.

Perhatikan 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)

```
12.00000000
```

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

>fx &= diff(xˆx,x); $&fx

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

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

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

```
@(a,b) a^2+b^2
41.00000000
```

Hal ini memungkinkan untuk memanipulasi ekspresi dalam variabel lain untuk fungsi EMT yang membutuhkan ekspresi dalam "x".

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

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

>fx &= xˆ3-a*x; . . .
> a=1.2; fx(0.5)

```
-0.47500000
```

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

>fx(0.5,a=1.1)

```
-0.42500000
```

Sebuah ekspresi tidak perlu berbentuk simbolik. Hal ini diperlukan, jika ekspresi mengandung fungsi-fungsi, yang hanya dikenal di kernel numerik, bukan di Maxima.

Contoh

>r:=7; fx:="pi*r^2"; longest fx()

```
      153.9380400258999
```

>r:=10; fx:="pi*r^2"; longest fx()

```
      314.1592653589793
```

>fx &= 1/2*a*x; ...
> a=1.2; fx(0.5)

```
 0.30000000
```

>fx(0.5,a=6)

```
 1.50000000
```

>fx(0.5,a=2)

```
 0.50000000
```

Matematika Simbolik

EMT melakukan matematika simbolik dengan bantuan Maxima. Untuk detailnya, mulailah dengan tutorial berikut ini, atau telusuri referensi untuk Maxima. Para ahli dalam Maxima harus memperhatikan bahwa ada perbedaan dalam sintaks antara sintaks asli Maxima dan sintaks default dari ekspresi simbolik dalam EMT.

Matematika simbolik diintegrasikan secara mulus ke dalam Euler dengan &. Ekspresi apapun yang dimulai dengan & adalah sebuah ekspresi simbolik. Ekspresi ini 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

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

$$2481256778$$

Tentu saja, Maxima memiliki fungsi yang lebih efisien untuk hal 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. Sebagai contoh, coba klik dua kali pada "&binomial" di baris perintah sebelumnya. Ini akan membuka dokumentasi Maxima yang disediakan oleh pembuat program tersebut.

Anda akan mengetahui bahwa fungsi-fungsi berikut juga dapat digunakan.

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

>$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)

$$120$$

Dengan begitu, Anda dapat menggunakan solusi dari sebuah persamaan dalam persamaan lain.

Ekspresi simbolik dicetak oleh Maxima dalam bentuk 2D. Alasannya adalah adanya sebuah bendera simbolik khusus pada string.

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

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

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

Ekspresi simbolik diuraikan oleh Euler. Jika Anda membutuhkan sintaks yang kompleks dalam satu ekspresi, Anda dapat mengapit ekspresi dalam "…". Menggunakan lebih dari satu ekspresi sederhana dimungkinkan, tetapi sangat tidak disarankan.

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

```
25
```

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

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

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

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

Sekali lagi, % mengacu pada hasil sebelumnya.

Untuk mempermudah, kita menyimpan solusi ke dalam sebuah 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))

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

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

>&factor(20!)

```
2432902008176640000
```

>::: factor(10!)

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

>:: factor(20!)

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

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

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

```
 2
g
```

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

```
 3  x
x  E
```

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

Such variables can be used in other symbolic expressions. Note, that in the following command the right hand side of &= is evaluated before the assignment to Fx.

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

```
     5
125 E
```

$$125\,e^5$$

18551.64488782208

>fx(5)

```
Real 1 x 101 matrix

390625.00000000 125.00000000 20131375.00000000 510512625.00000000     ...
```

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

Baris perintah berikut ini juga mendemonstrasikan bahwa Maxima dapat mengevaluasi sebuah 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 sebuah ekspresi, Anda dapat menggunakan perintah tex.

>tex(fx)

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

Ekspresi simbolik dapat dievaluasi seperti halnya ekspresi numerik.

>fx(0.5)

```
Real 1 x 101 matrix

 0.27950850  0.12500000  0.41457810  0.57282196     ...
```

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

>$&fx with x=1/2

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

Penugasan ini juga bisa bersifat simbolis.

>$&fx with x=1+t

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

Perintah solve menyelesaikan ekspresi simbolik untuk sebuah variabel di Maxima. Hasilnya adalah sebuah 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 "solve" numerik di Euler, yang membutuhkan nilai awal, dan secara opsional nilai target.

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

```
 1.56155281
```

Nilai numerik dari solusi simbolik dapat dihitung dengan evaluasi hasil simbolik. Euler akan membaca penugasan x= dst. Jika Anda tidak membutuhkan hasil numerik untuk perhitungan lebih lanjut, Anda juga bisa membiarkan Maxima menemukan nilai numeriknya.

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

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

-3.23606798 1.23606798

$$\left[x=-3.23606797749979,x=1.23606797749979\right]$$

Untuk mendapatkan solusi simbolik yang spesifik, seseorang dapat menggunakan “dengan” dan indeks.

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

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

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

Untuk menyelesaikan 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]

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

$$2$$

Ekspresi simbolik dapat memiliki bendera, yang menunjukkan perlakuan khusus di Maxima. Beberapa flag dapat digunakan sebagai perintah juga, namun ada juga yang tidak. Bendera 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(%)\

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

contoh soal

>$&24!

$$620448401733239439360000$$

>$& diff((4+3*a)/(3*a+4),a)

$$0$$

>$& diff((x*3-6*x$^{2+9*x)/(x}$3-3*x^2),x)

$$\frac{12-12\,x}{x^3-3\,x^2}-\frac{\left(12\,x-6\,x^2\right)\,\left(3\,x^2-6\,x\right)}{\left(x^3-3\,x^2\right)^2}$$

>$& diff((x$^{2+2*x-3)/(x}$2-9),x)

$$\frac{2\,x+2}{x^2-9}-\frac{2\,x\,\left(x^2+2\,x-3\right)}{\left(x^2-9\right)^2}$$

>$&solve(((x$^{2+2*x-3)/(x}$2-9)))

$$\left[x=1\right]$$

Fungsi

Dalam EMT, fungsi adalah program yang ditentukan dengan perintah “function”. Fungsi dapat berupa fungsi satu baris atau fungsi multibaris.

Fungsi satu baris dapat berupa numerik atau simbolik. Fungsi satu baris numerik didefinisikan dengan ":=".

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

Sebagai gambaran umum, kami menunjukkan semua definisi yang mungkin untuk fungsi satu baris. Sebuah fungsi dapat dievaluasi seperti halnya fungsi Euler bawaan.

>f(2)

```
 4.47213595
```

Fungsi ini juga dapat digunakan untuk vektor, mengikuti bahasa matriks Euler, karena ekspresi yang digunakan dalam fungsi ini adalah vektor.

>f(0:0.1:1)

```
Real 1 x 11 matrix

 0.00000000  0.10049876  0.20396078  0.31320920     ...
```

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

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

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

```
 0.78615138
```

Secara default, jika Anda perlu menimpa fungsi built-in, Anda harus menambahkan kata kunci "overwrite". Menimpa fungsi bawaan berbahaya dan dapat menyebabkan masalah bagi fungsi lain yang bergantung pada fungsi tersebut.

Anda masih dapat memanggil fungsi bawaan sebagai "_...", jika fungsi tersebut merupakan fungsi dalam inti Euler.

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

>sin(45)

```
 0.70710678
```

Sebaiknya kita hilangkan definisi ulang tentang dosa ini.

>forget sin; sin(pi/4)

```
 0.70710678
```

Contoh

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

akan dihitung nilai fungsi jika

>f(2)

```
 6.00000000
```

>f(sin(0))

```
 0.00000000
```

>f(2)+f(sin(0))

```
 6.00000000
```

>f(20)

```
420.00000000
```

>f(20)-10

```
410.00000000
```

Parameter Default

Fungsi numerik dapat memiliki parameter default.

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

Menghilangkan parameter ini menggunakan nilai default.

>f(4)

```
16.00000000
```

Menetapkannya akan menimpa nilai default.

>f(4,5)

```
80.00000000
```

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

>f(4,a=1)

```
16.00000000
```

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

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

>a=6; f(2)

```
24.00000000
```

Tetapi parameter yang ditetapkan akan menggantikan nilai global.

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

>f(2,a:=5)

```
20.00000000
```

Fungsi simbolik didefinisikan dengan "&=". Fungsi-fungsi ini didefinisikan dalam Euler dan Maxima, dan dapat digunakan di kedua bahasa tersebut. Ekspresi pendefinisian dijalankan melalui Maxima sebelum definisi.

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

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

Fungsi simbolis dapat digunakan dalam ekspresi simbolis.

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

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

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

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

>g(5+g(1))

```
Real 1 x 101 matrix

195.11148784 120.00000000 206.32982305 210.89785112     ...
```

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)

```
Cannot use vectors for conditions, use all(...)!
Maybe you need to vectorize the function with "map".
secant:
    if x2~=x1; break; endif;
Try "trace errors" to inspect local variables after errors.
solve:
    if eps then return secant(f$,a,b,y;args(),eps=eps);
```

Hal berikut ini juga dapat digunakan, karena Euler menggunakan ekspresi simbolik dalam fungsi g, jika tidak menemukan variabel simbolik g, dan jika ada fungsi simbolik g.

>solve($g,0.5)

```
Cannot use vectors for conditions, use all(...)!
Maybe you need to vectorize the function with "map".
secant:
    if x2~=x1; break; endif;
Try "trace errors" to inspect local variables after errors.
solve:
    if eps then return secant(f$,a,b,y;args(),eps=eps);
```

>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)

$$\left(x+2\right)^{n}$$

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

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

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

>P(3,4)

```
625.00000000
```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$$x^3-x$$

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

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

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

Dengan := fungsi tersebut berupa angka. Contoh yang baik adalah integral pasti seperti

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

yang tidak dapat dievaluasi secara simbolik.

Jika kita mendefinisikan ulang fungsi tersebut dengan kata kunci "map", maka 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)

```
-0.78343051 -0.41081565  0.00000000  0.67686328  2.05044623
```

Fungsi dapat memiliki nilai default untuk parameter.

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

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

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

```
 2.00000000
 6.70000000
```

Selain itu, dimungkinkan untuk menggunakan parameter yang ditetapkan.

>mylog(E^2,base=E)

```
Floating point error!
Try "trace errors" to inspect local variables after errors.
mylog:
    useglobal; return ln(x)/ln(base);
Error in:
mylog(E^2,base=E) ...
                ^
```

Sering kali, kita ingin menggunakan fungsi untuk vektor di satu tempat, dan untuk masing-masing elemen 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)

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

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

Fungsi simbolik seperti itu dapat digunakan untuk variabel simbolik.

Tetapi fungsi ini juga dapat digunakan untuk vektor numerik.

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

```
17.00000000
```

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

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

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

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

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

$$0$$

Tetapi tentu saja, semua itu bisa digunakan dalam ekspresi simbolis atau dalam definisi fungsi simbolis.

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

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

Untuk meringkas

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

Contoh

>function P(x,n) &= (x+1)^n; $&P(x,n)

$$(x+1)^n$$

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

$$(x+1)^4$$

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

>P(3,4)

```
256.00000000
```

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

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

$$x^4+3\,x^3-8\,x-7$$

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

$$(x+1)^4-(x+2)^3$$
$$x^4+3\,x^3-8\,x-7$$
$$x^4+3\,x^3-8\,x-7$$

Memecahkan Ekspresi

Ekspresi dapat diselesaikan secara numerik dan simbolik.

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

>solve(“x^2-2”,1)

```
1.41421356
```

Hal ini juga bisa digunakan untuk ekspresi simbolis. Perhatikan fungsi berikut ini.

>$&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])

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

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

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

Sekarang kita mencari titik, di mana polinomialnya adalah 2. Dalam solve(), nilai target default y=0 dapat diubah dengan variabel yang ditetapkan.

Kami menggunakan y=2 dan mengeceknya dengan mengevaluasi polinomial pada hasil sebelumnya.

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

```
0.96671559
2.00000000
```

Memecahkan sebuah ekspresi simbolik dalam bentuk simbolik mengembalikan sebuah 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])

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

$$0$$

Menyelesaikan sistem persamaan secara simbolik dapat dilakukan dengan vektor persamaan dan pemecah simbolik solve(). Jawabannya adalah daftar daftar persamaan Untuk menggunakan solusi secara simbolik dalam ekspresi lain, cara termudah adalah "dengan".

>$&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. Tetapi seringkali kita ingin menggunakan parameter lokal.

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

dengan a = 3.

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

Salah satu cara untuk mengoper parameter tambahan ke f() adalah dengan menggunakan sebuah daftar yang berisi nama fungsi dan parameternya (cara lainnya adalah dengan menggunakan parameter titik koma).

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

```
2.54116292
```

Hal ini juga dapat dilakukan dengan ekspresi. Namun, elemen daftar bernama harus digunakan. (Lebih lanjut tentang daftar dalam tutorial tentang sintaks EMT).

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

```
2.54116292
```

Contoh

>solve("x^{4-3*x}2+2",1)

```
1.00000000
```

>solve("x^{4+3*x}2-10",2)

```
1.41421356
```

>solve("5*x^2+3*x-1",1)

```
0.23851648
```

>solve("x^2-8*x+5",1)

```
0.68337521
```

>solve("x^2+4-5*x",1)

```
1.00000000
```

Menyelesaikan Pertidaksamaan

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

>&load(fourier_elim)

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

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

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

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

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

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

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

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

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

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

$$emptyset$$

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

$$universalset$$

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

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

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

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

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

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

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

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

>$&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 &lt; x, x &lt; 8, y &lt; - 11] or [8 &lt; x, y &lt; - 11]
 or [x &lt; 8, 13 &lt; y] or [x = y, 13 &lt; y] or [8 &lt; x, x &lt; y, 13 &lt; y]
 or [y &lt; x, 13 &lt; y]
```

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

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

contoh soal

>&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

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

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

$$[x < 1] \lor [1 < x]$$

>$&fourier_elim([x^3+1>0],[x]) // x^3+1>0

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

>$&fourier_elim([16*x^2>20],[x]) //16*x^2>20

$$\left[4\,x^2 - 5 > 0\right]$$

>$&fourier_elim([9*x^2+30*x+25<0],[x]) //9*x^2+30*x+25<0

$$emptyset$$

Bahasa Matrix

Dokumentasi inti EMT berisi diskusi 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]

```
 1.00000000  2.00000000
 3.00000000  4.00000000
```

Hasil kali matriks dilambangkan dengan sebuah titik.

>b=[3;4]

```
 3.00000000
 4.00000000
```

>b' // transpose b

```
 3.00000000  4.00000000
```

Transpose matriks adalah matriks baru yang diperoleh dengan menukar elemen baris menjadi kolom dan sebaliknya.

>inv(A) //inverse A

```
-2.00000000  1.00000000
 1.50000000 -0.50000000
```

Invers matriks adalah kebalikan (invers) dari sebuah matriks yang apabila matriks tersebut dikalikan dengan inversnya, akan menjadi matriks identitas.

>A.b //perkalian matriks

```
11.00000000
25.00000000
```

>A.inv(A)

```
 1.00000000  0.00000000
 0.00000000  1.00000000
```

Poin utama dari bahasa matriks adalah bahwa semua fungsi dan operator bekerja elemen demi elemen.

>A.A

```
 7.00000000 10.00000000
15.00000000 22.00000000
```

>A^2 //perpangkatan elemen2 A

```
 1.00000000  4.00000000
 9.00000000 16.00000000
```

A.A dengan A^2 menghasilkan dua matrix yang berbeda. Namun, pada perkalian matrix, formula yang benar adalah menggunakan A.A atau power(A,2). Hal ini dikarenakan formula A^2 hanya akan memangkatkan dua, tiap elemen-elemen dalam A.

>A.A.A

```
37.00000000 54.00000000
81.00000000 118.00000000
```

>power(A,3) //perpangkatan matriks

```
37.00000000 54.00000000
81.00000000 118.00000000
```

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

```
 1.00000000  1.00000000
 1.00000000  1.00000000
```

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

```
 0.33333333  0.66666667
 0.75000000  1.00000000
```

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

```
-2.00000000
 2.50000000
```

>inv(A).b

```
-2.00000000
 2.50000000
```

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

```
 1.00000000  0.00000000
 0.00000000  1.00000000
```

>inv(A).A

```
 1.00000000  0.00000000
 0.00000000  1.00000000
```

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

```
 1.00000000  4.00000000
 9.00000000 16.00000000
```

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

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

```
 9.00000000
16.00000000
```

Jika salah satu operan adalah vektor atau skalar, maka operan tersebut akan diperluas dengan cara alami.

>2*A

```
 2.00000000  4.00000000
 6.00000000  8.00000000
```

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

>[1,2]*A

```
 1.00000000  4.00000000
 3.00000000  8.00000000
```

Jika ini adalah vektor baris, vektor ini diterapkan ke semua kolom A.

>A*[2,3]

```
 2.00000000  6.00000000
 6.00000000 12.00000000
```

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

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

```
 1.00000000  2.00000000
 1.00000000  2.00000000
```

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

```
 1.00000000  4.00000000
 3.00000000  8.00000000
```

Hal ini juga berlaku untuk dua vektor di mana satu vektor adalah vektor baris dan yang lainnya adalah vektor kolom. Kami menghitung i*j untuk i, j dari 1 sampai 5. Caranya adalah dengan mengalikan 1:5 dengan transposenya. Bahasa matriks Euler secara otomatis menghasilkan sebuah tabel nilai.

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

```
 1.00000000  2.00000000  3.00000000  4.00000000  5.00000000
 2.00000000  4.00000000  6.00000000  8.00000000 10.00000000
 3.00000000  6.00000000  9.00000000 12.00000000 15.00000000
 4.00000000  8.00000000 12.00000000 16.00000000 20.00000000
 5.00000000 10.00000000 15.00000000 20.00000000 25.00000000
```

Sekali lagi, ingatlah bahwa ini bukan produk matriks!
>(1:5).(1:5)' // hasilkali vektor baris dan vektor kolom

```
55.00000000
```

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

```
55.00000000
```

Bahkan operator seperti < atau == bekerja dengan cara yang sama.
>(1:10)<6 // menguji elemen-elemen yang kurang dari 6

```
Real 1 x 10 matrix

 1.00000000  1.00000000  1.00000000  1.00000000     ...
```

Sebagai contoh, kita dapat menghitung jumlah elemen yang memenuhi kondisi tertentu dengan fungsi sum().
>sum((1:10)<6) // banyak elemen yang kurang dari 6

```
 5.00000000
```

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

```
Real 1 x 10 matrix

 0.00000000  0.00000000  0.00000000  0.00000000     ...
```

Dari vektor seperti itu, "nonzeros" memilih elemen bukan nol.
Dalam hal ini, kita mendapatkan indeks semua elemen yang lebih besar dari 50.
>nonzeros(t>50) //indeks elemen2 t yang lebih besar daripada 50

```
 8.00000000  9.00000000 10.00000000
```

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

```
64.00000000 81.00000000 100.00000000
```

Sebagai contoh, mari kita cari semua kuadrat dari angka 1 sampai 1000, yaitu 5 modulo 11 dan 3 modulo 13.
>t=1:1000; nonzeros(mod(t^2,11)==5 && mod(t^2,13)==3)

```
Real 1 x 28 matrix

 4.00000000 48.00000000 95.00000000 139.00000000     ...
```

EMT tidak sepenuhnya efektif untuk komputasi bilangan bulat. EMT menggunakan floating point presisi ganda secara internal. Akan tetapi, hal ini sering kali sangat berguna.

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

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

```
112.00000000
```

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

>seed(2); A=random(3,4)

```
 0.76576085  0.40118837  0.40634677  0.26782878
 0.13672954  0.39056680  0.49597500  0.95281420
 0.54813768  0.00608500  0.44425455  0.53924589
```

Ini mengembalikan indeks elemen, yang bukan nol.

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

```
 1.00000000  4.00000000
 2.00000000  1.00000000
 2.00000000  2.00000000
 3.00000000  2.00000000
```

Indeks ini dapat digunakan untuk menetapkan elemen ke suatu nilai.

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

```
 0.76576085  0.40118837  0.40634677  0.00000000
 0.00000000  0.00000000  0.49597500  0.95281420
 0.54813768  0.00000000  0.44425455  0.53924589
```

Fungsi mset() juga dapat mengatur elemen-elemen pada indeks ke entri-entri matriks lain.

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

```
 0.76576085  0.40118837  0.40634677 -0.12691737
-0.12240366 -0.69167311  0.49597500  0.95281420
 0.54813768 -0.48390213  0.44425455  0.53924589
```

Dan dimungkinkan untuk mendapatkan elemen-elemen dalam vektor.

>mget(A,k)

```
 0.26782878  0.13672954  0.39056680  0.00608500
```

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

>ex=extrema(A)

```
 0.26782878  4.00000000  0.76576085  1.00000000
 0.13672954  1.00000000  0.95281420  4.00000000
 0.00608500  2.00000000  0.54813768  1.00000000
```

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

>ex[,3]'

```
 0.76576085  0.95281420  0.54813768
```

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

>max(A)'

```
 0.76576085  0.95281420  0.54813768
```

Tetapi dengan mget(), kita dapat mengekstrak indeks dan menggunakan informasi ini untuk mengekstrak elemen-elemen pada posisi yang sama dari matriks yang lain.

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

```
 1.00000000  1.00000000
 2.00000000  4.00000000
 3.00000000  1.00000000
-0.76576085 -0.95281420 -0.54813768
```

Contoh

>B=[1,1;2,4]

```
 1.00000000  1.00000000
 2.00000000  4.00000000
```

>B*B

```
 1.00000000  1.00000000
 4.00000000 16.00000000
```

>B^2

```
 1.00000000  1.00000000
 4.00000000 16.00000000
```

>inv(B)

```
 2.00000000 -0.50000000
-1.00000000  0.50000000
```

>B\B

```
 1.00000000  0.00000000
 0.00000000  1.00000000
```

>2*B

```
 2.00000000  2.00000000
 4.00000000  8.00000000
```

Fungsi Matriks Lainnya (Membangun Matriks)

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

>v=1:3; v_v

```
 1.00000000  2.00000000  3.00000000
 1.00000000  2.00000000  3.00000000
```

Demikian juga, kita dapat melampirkan matriks ke matriks lain secara berdampingan, jika keduanya memiliki jumlah baris yang sama.

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

```
 0.03244399  0.05341706  0.59571341  0.56445382  1.00000000
 0.83915970  0.17555216  0.39698834  0.83514023  2.00000000
 0.02575733  0.65858524  0.62983210  0.77089513  3.00000000
```

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

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

>A|1

```
 0.03244399  0.05341706  0.59571341  0.56445382  1.00000000
 0.83915970  0.17555216  0.39698834  0.83514023  1.00000000
 0.02575733  0.65858524  0.62983210  0.77089513  1.00000000
```

Dimungkinkan untuk membuat matriks vektor baris dan kolom.

>[v;v]

```
 1.00000000  2.00000000  3.00000000
 1.00000000  2.00000000  3.00000000
```

>[v',v']

```
 1.00000000  1.00000000
 2.00000000  2.00000000
 3.00000000  3.00000000
```

Tujuan utama dari hal ini adalah untuk menginterpretasikan vektor ekspresi untuk vektor kolom.

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

```
 1.00000000  1.00000000
 2.00000000  4.00000000
 3.00000000  9.00000000
```

Untuk mendapatkan ukuran A, kita dapat menggunakan fungsi berikut ini.

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

```
 2.00000000
 4.00000000
 2.00000000  4.00000000
 4.00000000
```

Untuk vektor, ada length().

>length(2:10)

```
 9.00000000
```

Ada banyak fungsi lain yang menghasilkan matriks.

>ones(2,2)

```
 1.00000000  1.00000000
 1.00000000  1.00000000
```

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

>ones(5)*6

```
 6.00000000  6.00000000  6.00000000  6.00000000  6.00000000
```

Matriks angka acak juga dapat dibuat dengan acak (distribusi seragam) atau normal (distribusi Gauß).

>random(2,2)

```
 0.66566020  0.83183501
 0.97699964  0.54425848
```

Here is another useful function, which restructures the elements of a matrix into another matrix.

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

```
 1.00000000  2.00000000  3.00000000
 4.00000000  5.00000000  6.00000000
 7.00000000  8.00000000  9.00000000
```

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

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

Mari kita uji.

>rep(1:3,5)

```
Real 1 x 15 matrix

 1.00000000  2.00000000  3.00000000  1.00000000     ...
```

Fungsi multdup() menduplikasi elemen-elemen sebuah vektor.

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

```
Real 1 x 15 matrix

 1.00000000  1.00000000  1.00000000  1.00000000     ...
Real 1 x 7 matrix

 1.00000000  1.00000000  2.00000000  2.00000000     ...
```

Fungsi flipx() dan flipy() membalik urutan baris atau kolom dari sebuah matriks. Misalnya, fungsi flipx() membalik secara horizontal.

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

```
 5.00000000  4.00000000  3.00000000  2.00000000  1.00000000
```

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

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

```
 2.00000000  3.00000000  4.00000000  5.00000000  1.00000000
```

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

>drop(10:20,3)

```
Real 1 x 10 matrix

10.00000000 11.00000000 13.00000000 14.00000000     ...
```

Perhatikan bahwa vektor i dalam drop(v,i) merujuk pada indeks elemen dalam v, bukan nilai elemen. Jika Anda ingin menghapus elemen, Anda harus menemukan elemen-elemen tersebut terlebih dahulu. Fungsi indexof(v,x) dapat digunakan untuk menemukan elemen x dalam vektor terurut v.

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

```
Real 1 x 15 matrix

 2.00000000  3.00000000  5.00000000  7.00000000     ...
Real 1 x 11 matrix

 0.00000000  5.00000000  0.00000000  6.00000000     ...
Real 1 x 11 matrix

 2.00000000  3.00000000  5.00000000  7.00000000     ...
```

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

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

```
Real 1 x 8 matrix

 1.00000000  2.00000000  3.00000000  4.00000000     ...
```

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

Kita mulai dengan matriks identitas.

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

```
 1.00000000  0.00000000  0.00000000  0.00000000  0.00000000
 0.00000000  1.00000000  0.00000000  0.00000000  0.00000000
 0.00000000  0.00000000  1.00000000  0.00000000  0.00000000
 0.00000000  0.00000000  0.00000000  1.00000000  0.00000000
 0.00000000  0.00000000  0.00000000  0.00000000  1.00000000
```

Kemudian, kami menetapkan diagonal bawah (-1) ke 1:4.

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

```
 1.00000000  0.00000000  0.00000000  0.00000000  0.00000000
 1.00000000  1.00000000  0.00000000  0.00000000  0.00000000
 0.00000000  2.00000000  1.00000000  0.00000000  0.00000000
 0.00000000  0.00000000  3.00000000  1.00000000  0.00000000
 0.00000000  0.00000000  0.00000000  4.00000000  1.00000000
```

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

Berikut adalah sebuah fungsi yang mengembalikan sebuah matriks tri-diagonal.

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

```
 2.00000000  3.00000000  0.00000000  0.00000000  0.00000000
 1.00000000  2.00000000  3.00000000  0.00000000  0.00000000
 0.00000000  1.00000000  2.00000000  3.00000000  0.00000000
 0.00000000  0.00000000  1.00000000  2.00000000  3.00000000
 0.00000000  0.00000000  0.00000000  1.00000000  2.00000000
```

Diagonal sebuah matriks juga dapat diekstrak dari matriks. Untuk mendemonstrasikan hal ini, kami merestrukturisasi vektor 1:9 menjadi matriks 3x3.

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

```
 1.00000000  2.00000000  3.00000000
 4.00000000  5.00000000  6.00000000
 7.00000000  8.00000000  9.00000000
```

Sekarang kita bisa mengekstrak diagonal.

>d=getdiag(A,0)

```
 1.00000000  5.00000000  9.00000000
```

Contoh: Kita dapat membagi matriks dengan diagonalnya. Bahasa matriks memperhatikan 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
```

Contoh

>v=2:4; v_v

```
 2.00000000  3.00000000  4.00000000
 2.00000000  3.00000000  4.00000000
```

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

```
 0.20856566  0.22014429  0.85539891  0.02885462  2.00000000
 0.25928609  0.18137882  0.29364238  0.79149686  3.00000000
 0.01550552  0.31275422  0.38138684  0.87538050  4.00000000
```

>B|1

```
 0.20856566  0.22014429  0.85539891  0.02885462  1.00000000
 0.25928609  0.18137882  0.29364238  0.79149686  1.00000000
 0.01550552  0.31275422  0.38138684  0.87538050  1.00000000
```

>[v;v]

```
 2.00000000  3.00000000  4.00000000
 2.00000000  3.00000000  4.00000000
```

>[v',v']

```
 2.00000000  2.00000000
 3.00000000  3.00000000
 4.00000000  4.00000000
```

Vektorisasi

Hampir semua fungsi dalam Euler juga dapat digunakan untuk input matriks dan vektor, jika hal ini masuk akal.

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

>sqrt(1:3)

```
 1.00000000  1.41421356  1.73205081
```

Jadi, Anda dapat dengan mudah membuat tabel nilai. Ini adalah salah satu cara untuk memplot sebuah fungsi (alternatif lainnya 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 dihasilkan dengan mudah.

Pada contoh berikut, kita membuat vektor nilai t[i] dengan jarak 0.1 dari -1 hingga 1. Kemudian kita membuat vektor nilai dari fungsi

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

```
Real 1 x 21 matrix

 0.00000000  0.17100000  0.28800000  0.35700000     ...
```

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

Misalnya, vektor kolom dikalikan vektor baris diperluas 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 dengan hasil kali matriks. Hasil kali matriks dilambangkan dengan sebuah titik "." dalam EMT.

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

```
55.00000000
```

Secara default, vektor baris dicetak dalam format ringkas.

>[1,2,3,4]

```
 1.00000000  2.00000000  3.00000000  4.00000000
```

Untuk matriks, operator khusus . menyatakan perkalian matriks, dan A' menyatakan transposisi. Matriks 1x1 dapat digunakan seperti halnya bilangan real.

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

```
 5.00000000
25.00000000
```

Untuk mentransposisikan matriks, kita menggunakan apostrof.

>v=1:4; v'

```
 1.00000000
 2.00000000
 3.00000000
 4.00000000
```

Jadi kita dapat menghitung matriks A dikali vektor b.

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

```
30.00000000
70.00000000
```

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

>v'.v

```
 1.00000000  2.00000000  3.00000000  4.00000000
 2.00000000  4.00000000  6.00000000  8.00000000
 3.00000000  6.00000000  9.00000000 12.00000000
 4.00000000  8.00000000 12.00000000 16.00000000
```

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

>v.v'

```
30.00000000
```

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

>norm(v)^2

```
30.00000000
```

Operator dan fungsi mematuhi bahasa matriks Euler.

Berikut ini adalah ringkasan aturannya.

- Sebuah fungsi yang diterapkan pada vektor atau matriks diterapkan
- pada setiap elemen.
- Operator yang beroperasi pada dua matriks dengan ukuran yang sama
- diterapkan secara berpasangan pada elemen-elemen matriks.
- Jika dua matriks memiliki dimensi yang berbeda, keduanya diperluas
- dengan cara yang masuk akal, sehingga memiliki ukuran yang sama.

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

Berikut ini adalah kasus sederhana dengan operator ^.

>[1,2,3]^2

```
 1.00000000  4.00000000  9.00000000
```

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

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

```
 1.00000000  2.00000000  3.00000000
 2.00000000  4.00000000  6.00000000
 3.00000000  6.00000000  9.00000000
```

Perhatikan bahwa hasil kali skalar menggunakan hasil kali matriks, bukan tanda *!

>v.v'

```
14.00000000
```

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

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

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

```
 1.00000000  4.00000000  9.00000000
14.00000000
 1.00000000  5.00000000 14.00000000
```

Operator : menghasilkan vektor baris dengan spasi yang sama, opsional dengan ukuran langkah.

>1:4, 1:2:10

```
 1.00000000  2.00000000  3.00000000  4.00000000
 1.00000000  3.00000000  5.00000000  7.00000000  9.00000000
```

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

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

```
 1.00000000  2.00000000  3.00000000  4.00000000  5.00000000
 1.00000000  2.00000000  3.00000000
 1.00000000  1.00000000  1.00000000
```

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

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

```
 6.00000000
```

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

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

```
 6.00000000
 7.00000000  8.00000000  9.00000000
```

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

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

```
 2.00000000  4.00000000
 2.00000000
 5.00000000
 8.00000000
```

Bentuk singkat untuk : adalah menghilangkan indeks sepenuhnya.

>A[,2:3]

```
 2.00000000  3.00000000
 5.00000000  6.00000000
 8.00000000  9.00000000
```

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

>A{4}

```
 4.00000000
```

Sebuah matriks juga dapat diratakan, dengan menggunakan fungsi redim(). Hal ini diimplementasikan dalam fungsi flatten().

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

```
Real 1 x 9 matrix

 1.00000000  2.00000000  3.00000000  4.00000000     ...
Real 1 x 9 matrix

 1.00000000  2.00000000  3.00000000  4.00000000     ...
```

Untuk menggunakan matriks untuk tabel, mari kita atur ulang ke format default, dan menghitung tabel nilai sinus dan kosinus. Perhatikan bahwa sudut 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 membuat beberapa tabel dari beberapa fungsi sekaligus.

Pada contoh berikut, kita menghitung t[j]^i untuk i dari 1 hingga n. Kita mendapatkan sebuah matriks, di mana setiap baris adalah tabel t^i untuk satu i. Dengan kata lain, matriks tersebut memiliki elemen-elemen lateks: a_{i, j} = t_j^i, $1 \leq j \leq 101, 1 \leq i \leq n$

Sebuah fungsi yang tidak bekerja untuk input vektor harus "divektorkan". Hal ini dapat dicapai dengan kata kunci "map" dalam definisi fungsi. Kemudian fungsi akan dievaluasi untuk setiap elemen parameter vektor.

Integrasi numerik integrate() hanya bekerja untuk batas interval skalar. Jadi kita perlu membuat vektornya.

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

Kata kunci “map” membuat vektor fungsi. Fungsi ini sekarang akan bekerja untuk vektor angka.
>f([1:5])

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

Contoh

>f([1:5])

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

>sqrt(1:5)

```
[1,  1.41421,  1.73205,  2,  2.23607]
```

>A{8}

```
8
```

>v:=[2,4,7,8]; v[3], A[3]

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

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

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

Sub-Matriks dan Elemen Matriks

Untuk mengakses elemen matriks, gunakan notasi kurung.
>A=[1,2,3;4,5,6;7,8,9], A[2,2]

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

Kita dapat mengakses baris lengkap dari sebuah matriks.
>A[2]

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

Untuk 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 indeks, Euler akan mengembalikan baris-baris yang sesuai dari matriks.

Di sini kita menginginkan baris pertama dan kedua dari A.

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

Kita bahkan dapat menyusun ulang A menggunakan vektor indeks. Tepatnya, kita tidak mengubah A di sini, tetapi menghitung versi susunan ulang dari A.

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

Trik indeks juga bekerja 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
```

Sebagai alternatif, 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-elemen dari A dengan memberikan sebuah submatriks dari A ke suatu nilai. Hal ini sebenarnya mengubah matriks A yang tersimpan.

>A[1,1]=4

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

Kita juga dapat menetapkan nilai pada deretan A.

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

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

Kami bahkan dapat menetapkan ke sub-matriks jika memiliki ukuran yang tepat.

>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 akan mengembalikan matriks kosong, atau pesan kesalahan, tergantung pada pengaturan sistem. Standarnya adalah pesan kesalahan. Namun, ingatlah bahwa indeks negatif dapat digunakan untuk mengakses elemen-elemen matriks yang dihitung dari akhir.

>A[4]

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

Mengurutkan dan Mengacak

Fungsi sort() mengurutkan vektor baris.

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

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

Sering kali diperlukan untuk mengetahui indeks vektor yang diurutkan dalam vektor aslinya. Hal ini dapat digunakan untuk menyusun ulang vektor lain dengan cara yang sama.

Mari kita mengacak sebuah vektor.

>v=shuffle(1:10)

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

Indeks berisi urutan v yang tepat.

>{vs,ind}=sort(v); v[ind]

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

Hal ini juga berlaku 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 unique mengembalikan daftar terurut dari elemen unik sebuah vektor.

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

```
[5,  8,  5,  2,  7,  10,  4,  4,  2,  1]
[1,  2,  4,  5,  7,  8,  10]
```

Hal ini juga berlaku untuk vektor string.

>unique(s)

```
a
aa
d
e
```

Aljabar Linier

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

Untuk sistem linier Ax=b, Anda dapat menggunakan algoritma Gauss, matriks invers, atau kecocokan linier. Operator Amenggunakan versi algoritma Gauss.

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

```
          -4
         4.5
```

Sebagai contoh lain, kita membuat matriks 200x200 dan jumlah barisnya. Kemudian kita selesaikan Ax = b dengan menggunakan matriks kebalikannya. Kita mengukur kesalahan sebagai deviasi maksimal dari semua elemen dari 1, yang tentu saja merupakan solusi yang benar.

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

```
  8.810729923425242e-13
```

Jika sistem tidak memiliki 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
```

Determinan dari matriks ini adalah 0.

>det(A)

```
0
```

Matriks Simbolik

Maxima memiliki matriks simbolik. Tentu saja, Maxima dapat digunakan untuk masalah aljabar linier sederhana. Kita bisa mendefinisikan matriks untuk Euler dan Maxima dengan &:=, dan kemudian menggunakannya dalam ekspresi simbolik. Bentuk [. . .] yang biasa untuk mendefinisikan matriks dapat digunakan dalam 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(%)

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

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

>$&invert(A) with a=0

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

>A &= [1,a;b,2]; $A

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

Seperti semua variabel simbolik, matriks ini dapat digunakan dalam ekspresi simbolik lainnya.

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

$$\left(1-x\right)\,\left(2-x\right)-a\,b$$

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

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

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

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

Untuk mengekstrak vektor eigen tertentu, diperlukan pengindeksan yang cermat.

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

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

[1, - 1]

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

>A(a=4,b=5)

```
1                  4
5                  2
```

Dalam ekspresi simbolis, gunakan dengan.

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

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

Akses ke baris matriks simbolik bekerja seperti halnya matriks numerik.

>$&A[1]

$$[1, a]$$

Ekspresi simbolik dapat berisi sebuah penugasan. Dan itu mengubah matriks A.

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

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

Terdapat fungsi-fungsi simbolik dalam Maxima untuk membuat vektor dan matriks. Untuk hal ini, lihat dokumentasi Maxima atau tutorial tentang Maxima di EMT.

>v &= makelist(1/(i+j),i,1,3); $v

$$\left[\frac{1}{j+1}, \frac{1}{j+2}, \frac{1}{j+3}\right]$$

>B &:= [1,2;3,4]; $B, $&invert(B)

$$\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}$$

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

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

>$&invert(B)()

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

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

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

>longest B.xinv(B)

```
                      1                           0
                      0                           1
```

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 mengetahui detailnya.

>$&eigenvalues(**(A?)**)

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

Nilai Numerik dalam Ekspresi simbolik

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

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

```
            1         3.14159
            4               5
```

Masih ada perbedaan antara bentuk numerik dan bentuk simbolik. Ketika mentransfer matriks ke bentuk simbolik, perkiraan pecahan untuk bilangan real akan digunakan.

>$&A

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

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

>mxmset(A); $&A

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

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

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

$$1.4142135623730950488016887242097_B \times 10^0$$

$$1.414213562373095$$

Ketepatan angka floating point yang besar dapat diubah.

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

```
        3.14159265358979323846264338327950288419716939937510582097494\
459230781640628620899862803482534211706 8b0
```

Variabel numerik dapat digunakan dalam ekspresi simbolik apa pun dengan 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 5000 (katakanlah dalam dolar).

>K=5000

```
5000
```

Sekarang kita asumsikan suku bunga 3% per tahun. Mari kita tambahkan satu suku bunga sederhana dan hitung hasilnya.

>K*1.03

```
5150
```

Euler juga akan memahami sintaks berikut ini.

>K+K*3%

```
5150
```

Tetapi lebih mudah untuk menggunakan faktor

>q=1+3%, K*q

```
1.03
5150
```

Untuk 10 tahun, kita cukup mengalikan faktor-faktor tersebut dan mendapatkan nilai akhir dengan suku bunga majemuk.

>K*qˆ10

```
6719.58189672
```

Untuk tujuan kita, kita bisa menetapkan formatnya menjadi 2 digit setelah titik desimal.

>format(12,2); K*qˆ10

```
    6719.58
```

Mari kita cetak angka yang dibulatkan menjadi 2 digit dalam kalimat lengkap.

>"Starting from" + K + "$ you get" + round(K*qˆ10,2) + "$."

```
Starting from 5000$ you get 6719.58$.
```

Bagaimana jika kita ingin mengetahui hasil antara dari tahun ke-1 hingga tahun ke-9? Untuk hal ini, bahasa matriks Euler sangat membantu. Anda tidak perlu menulis perulangan, tetapi cukup masukkan

>K*qˆ(0:10)

```
Real 1 x 11 matrix

    5000.00     5150.00     5304.50     5463.64     ...
```

Bagaimana keajaiban ini bekerja? Pertama, ekspresi 0:10 mengembalikan sebuah vektor bilangan bulat.

>short 0:10

```
[0,  1,  2,  3,  4,  5,  6,  7,  8,  9,  10]
```

Kemudian semua operator dan fungsi dalam Euler dapat diterapkan pada vektor elemen demi elemen. Jadi

>short qˆ(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 qˆ0 hingga qˆ10. Ini dikalikan dengan K, dan kita mendapatkan vektor nilai.

>VK=K*qˆ(0:10);

Tentu saja, cara yang realistis untuk menghitung suku bunga ini adalah dengan membulatkan ke sen terdekat setelah setiap tahun. 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 mengulang selama bertahun-tahun. Euler menyediakan banyak solusi untuk ini.

Cara termudah adalah iterasi fungsi, yang mengulang fungsi yang diberikan beberapa kali.

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

```
Real 1 x 11 matrix

    5000.00      5150.00      5304.50      5463.64      ...
```

Kita bisa mencetaknya dengan cara yang bersahabat, menggunakan format kami dengan angka desimal yang tetap.

>VKr'

```
    5000.00
    5150.00
    5304.50
    5463.64
    5627.55
    5796.38
    5970.27
    6149.38
    6333.86
    6523.88
    6719.60
```

Untuk mendapatkan elemen tertentu dari vektor, kita menggunakan indeks dalam tanda kurung siku.

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

```
    5150.00
    5000.00      5150.00      5304.50
```

Yang mengejutkan, kita juga dapat 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.

Menyelesaikan Persamaan

Sekarang kita ambil fungsi yang lebih maju, yang menambahkan tingkat uang tertentu setiap tahun.

>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 bahwa uangnya berkurang. Jelas, jika kita hanya mendapatkan 150 bunga di tahun pertama, tetapi menghapus 200, kita kehilangan uang setiap tahun.

Bagaimana kita dapat menentukan berapa tahun uang itu akan bertahan? Kita harus menulis perulangan untuk ini. Cara termudah adalah dengan melakukan perulangan yang 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 karena nonzeros(VKR<0) mengembalikan vektor dengan indeks i, di mana 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. Fungsi ini dapat menerima kondisi akhir sebagai argumen. Kemudian fungsi ini 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. Anggaplah kita tahu bahwa nilainya adalah 0 setelah 50 tahun. Berapakah tingkat suku bunganya?

Ini adalah pertanyaan yang hanya bisa dijawab secara numerik. Di bawah ini, kami akan menurunkan rumus yang diperlukan. Kemudian Anda akan melihat bahwa tidak ada rumus yang mudah untuk suku bunga. Namun untuk saat ini, kita akan mencari solusi numerik.

Langkah pertama adalah mendefinisikan sebuah fungsi yang melakukan iterasi sebanyak n kali. Kita tambahkan semua parameter ke fungsi ini.

>function f(K,R,P,n) := iterate("x*(1+P/100)+R",K,n;P,R)[-1]

Perulangannya sama seperti di atas

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

etapi kita tidak lagi menggunakan nilai global R dalam ekspresi kita. Fungsi-fungsi seperti iterate() memiliki trik khusus dalam Euler. Anda bisa mengoper nilai variabel dalam ekspresi sebagai parameter titik koma. Dalam hal ini P dan R.

Selain itu, kita hanya tertarik pada nilai terakhir. Jadi kita mengambil indeks [-1].

Mari kita coba sebuah tes.

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

```
     -19.83
```

Sekarang kita bisa menyelesaikan masalah kita.

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

```
3.15
```

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

Kita dapat menggunakan fungsi yang sama untuk menyelesaikan pertanyaan berikut: Berapa banyak yang dapat kita hapus 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 Simbolik untuk Masalah Suku Bunga

Kita dapat menggunakan bagian simbolik dari Euler untuk mempelajari masalah ini. Pertama, kita mendefinisikan fungsi onepay() secara simbolik.

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

$$R + q\,K$$

Sekarang kita bisa mengulangi hal ini.

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

$$q\,\left(q\,\left(q\,\left(R + q\,K\right) + R\right) + R\right) + R$$

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

Kita melihat sebuah pola. Setelah n periode, kita memiliki

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

Rumus tersebut adalah rumus untuk jumlah geometris, yang dikenal dengan Maxima.

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

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

Ini sedikit rumit. Penjumlahan dievaluasi dengan flag "simpsum" untuk menguranginya menjadi hasil bagi.

Mari kita buat sebuah fungsi untuk ini.

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

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

Fungsi ini melakukan hal yang sama seperti fungsi f kita sebelumnya. Tetapi fungsi 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 kita adalah 30 tahun.

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

```
20.51
```

Jawaban ini mengatakan bahwa nilai tersebut akan menjadi negatif setelah 21 tahun.

Kita juga bisa menggunakan sisi simbolis dari Euler untuk menghitung rumus pembayaran.

Asumsikan kita mendapatkan pinjaman sebesar K, dan membayar n kali pembayaran sebesar R (dimulai setelah tahun pertama) sehingga menyisakan sisa utang sebesar Kn (pada saat pembayaran terakhir). Rumus untuk hal ini adalah sebagai berikut.

>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 laju 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 dapat Anda lihat dari rumusnya, fungsi ini mengembalikan kesalahan floating point untuk i = 0. Euler tetap memplotnya.

Tentu saja, kita memiliki batas berikut.

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

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

Jelasnya, tanpa bunga kita harus membayar kembali 10 suku bunga 500.

Persamaan ini juga dapat diselesaikan untuk n. Akan terlihat lebih baik jika kita menerapkan beberapa penyederhanaan.

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

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

unicodehyperref hyphensurl [ ]book xcolor amsmath,amssymb iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype [protrusion]basicmath parskip graphicx bookmark xurl same hidelinks, pdfcreator=LaTeX via pandoc

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 beberapa fungsi plot yang sangat mendasar. Terdapat koordinat layar, yang selalu berkisar dari 0 hingga 1024 di setiap sumbu, tidak peduli apakah layarnya persegi atau tidak. Semut ada koordinat plot, yang dapat diatur dengan setplot(). Pemetaan antara koordinat tergantung pada jendela plot saat ini. Sebagai contoh, default shrinkwindow() menyisakan ruang untuk label sumbu dan judul plot.

Pada contoh, kita hanya menggambar beberapa garis acak dalam berbagai warna. Untuk detail mengenai fungsi-fungsi ini, pelajari fungsi-fungsi inti dari 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); // mendapatkan poin acak

>colors=rgb(random(n),random(n),random(n)); // dapatkan warna acak

>loop 1 to n; color(colors[#]); plot(X[#],Y[#]); end; // plot

>hold off; // akhiri overwrite mode

>insimg; // masukan ke buku catatan

>reset;

Anda harus menahan grafik, karena perintah plot() akan menghapus jendela plot.

Untuk menghapus semua yang telah kita lakukan, kita gunakan reset().

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

For another example, we draw a plot as an inset in another plot. This is done by defining a smaller plot window. Note that this window does not provide room for the axis labels outside the plot window. We have to add some margin for this as needed. Note that we save and restore the full window, and hold the current plot while we plot the inset.

>plot2d("x^3-x");

>xw=200; yw=100; ww=300; hw=300;

>ow=window();

>window(xw,yw,xw+ww,yw+hw);

Gambar 1: images/PLOT%202D_Diva%20Nagita_23030630024-001.png

>hold on;

>barclear(xw-50,yw-10,ww+60,ww+60);

>plot2d("x^4-x",grid=6):

>hold off;

>window(ow);

Plot dengan beberapa angka dicapai dengan cara yang sama. Ada fungsi utility figure() untuk ini.

Aspek Plot

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

Tetapi Anda juga dapat mengubahnya untuk satu plot. Untuk ini, ukuran area plot saat ini diubah, dan jendela diatur sedemikian rupa sehingga label memiliki ruang yang cukup.

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

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

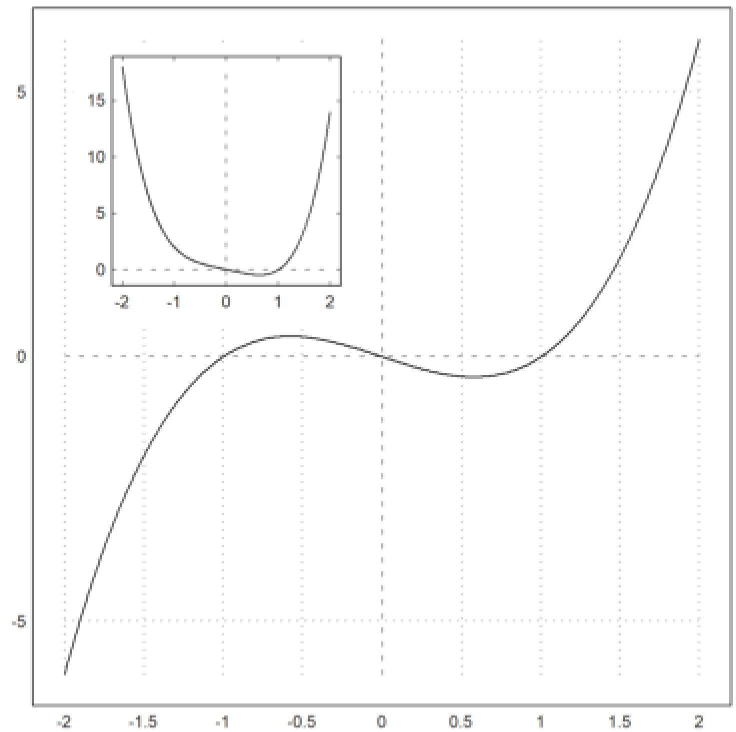
>aspect();

>reset;

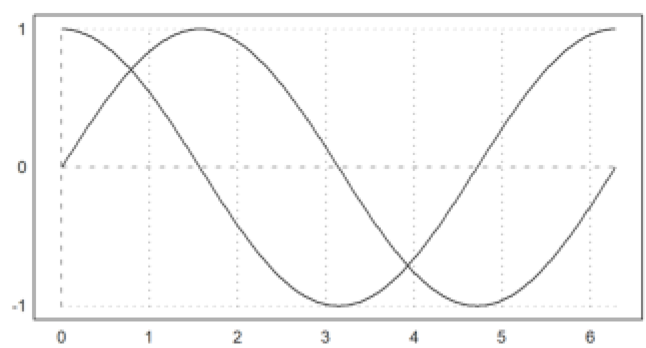
Fungsi reset () mengembalikan default plot, termasuk rasio aspek.

Plot 2D di Euler

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



Gambar 2: images/PLOT%202D_Diva%20Nagita_23030630024-002.png



Gambar 3: images/PLOT%202D_Diva%20Nagita_23030630024-003.png

Dimungkinkan untuk memplot di Maxima menggunakan Gnuplot atau di Python menggunakan Math Plot Lib.

Euler dapat memplot plot 2D dari

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

Gaya plot mencakup berbagai gaya untuk garis dan titik, plot batang, dan plot berbayang.

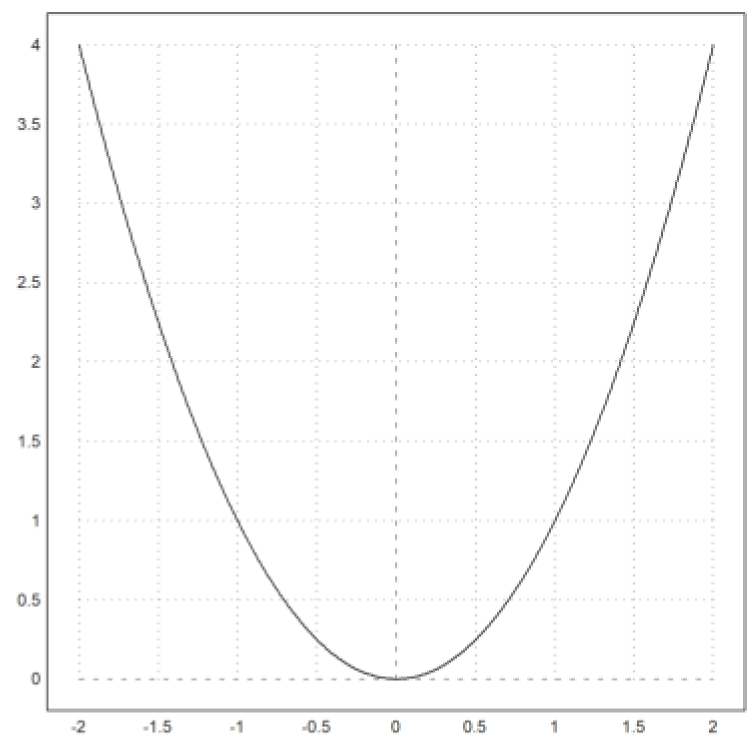
Plot Ekspresi atau Variabel

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

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

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

>plot2d("x^2"):



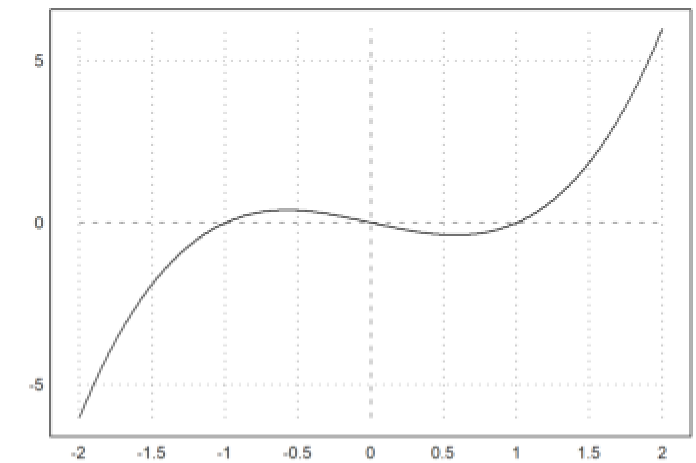
Gambar 4: images/PLOT%202D_Diva%20Nagita_23030630024-004.png

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

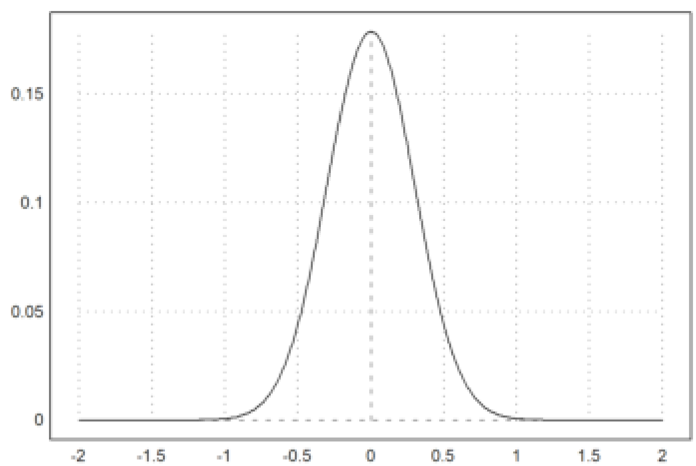
>a:=5.6; plot2d("exp(-a*x^2)/a"); insimg(30); // menampilkan gambar hasil plot setinggi 25 baris

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 ditetapkan dengan parameter yang ditetapkan berikut ini



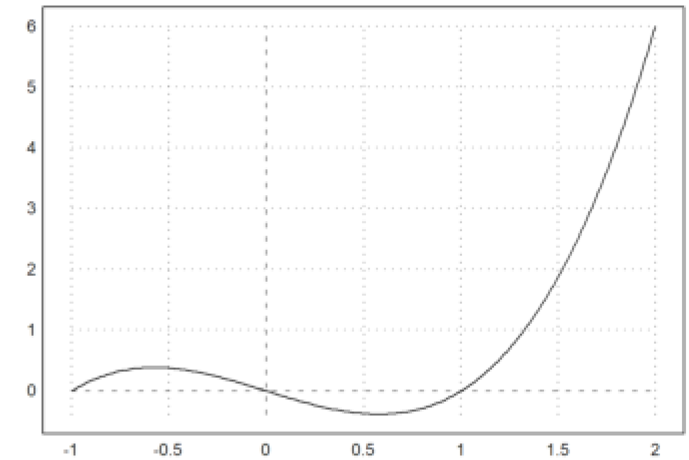
Gambar 5: images/PLOT%202D_Diva%20Nagita_23030630024-005.png



Gambar 6: images/PLOT%202D_Diva%20Nagita_23030630024-006.png

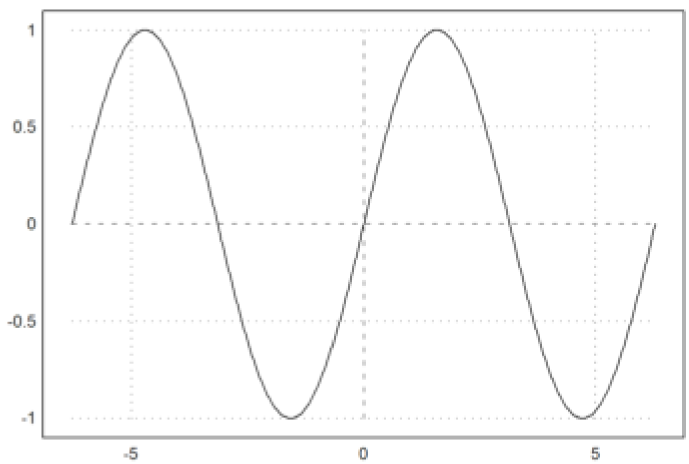
- a,b: rentang x (default -2,2)
- c, d: rentang y (default: skala dengan nilai)
- r: sebagai alternatif, radius di sekitar pusat plot
- cx, cy: koordinat pusat plot (default 0,0)

>plot2d("xˆ3-x",-1,2):



Gambar 7: images/PLOT%202D_Diva%20Nagita_23030630024-007.png

>plot2d("sin(x)",-2*pi,2*pi): // plot sin(x) pada interval [-2pi, 2pi]



Gambar 8: images/PLOT%202D_Diva%20Nagita_23030630024-008.png

>plot2d("cos(x)","sin(3*x)",xmin=0,xmax=2pi):

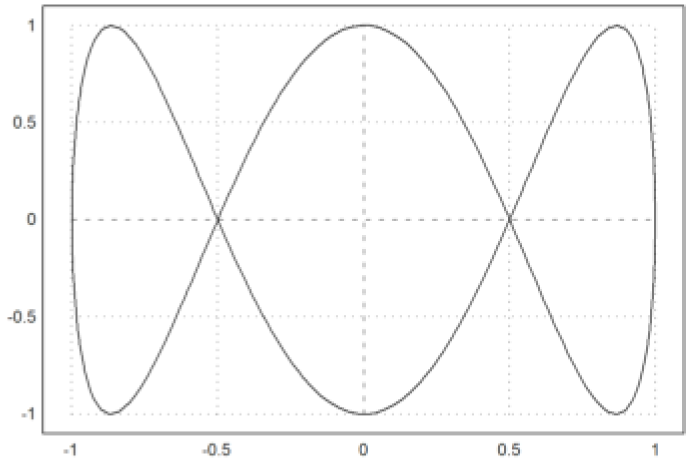
Alternatif untuk tanda titik dua adalah perintah insimg(lines), yang menyisipkan plot yang menempati sejumlah baris teks tertentu.

Dalam opsi, plot dapat diatur untuk muncul

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

Lebih banyak gaya yang dapat dicapai dengan perintah plot tertentu.

Dalam hal apa pun, tekan tombol tabulator untuk melihat plot, jika disembunyikan.

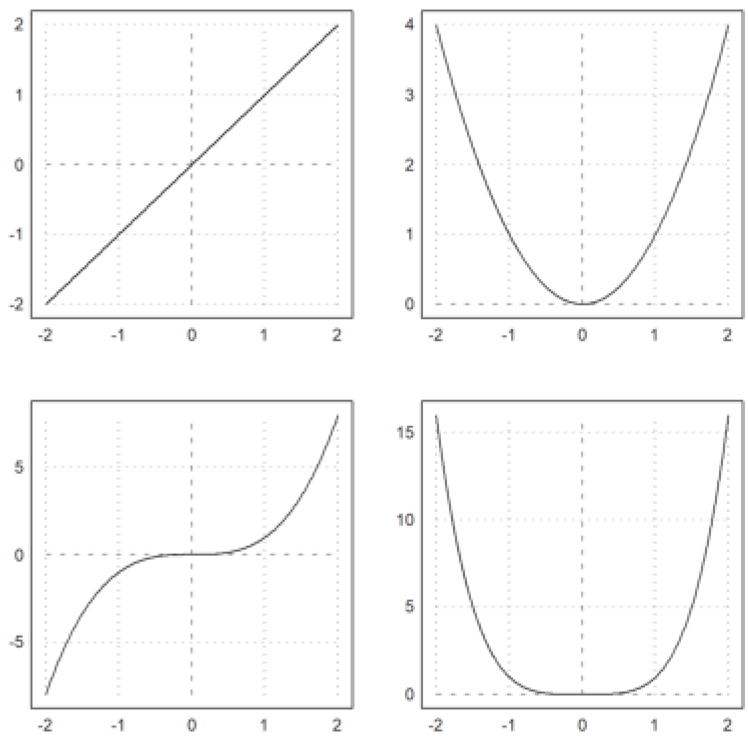


Gambar 9: images/PLOT%202D_Diva%20Nagita_23030630024-009.png

Untuk membagi jendela menjadi beberapa plot, gunakan perintah figure(). Pada contoh, kita memplot x^1 sampai x^4 ke dalam 4 bagian jendela. figure(0) akan mereset jendela default.

>reset;

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

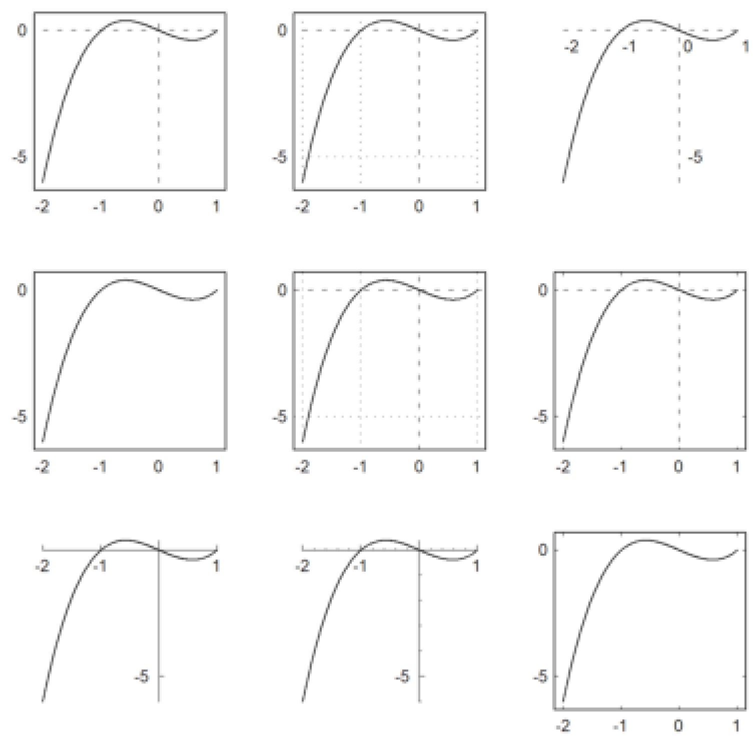


Gambar 10: images/PLOT%202D_Diva%20Nagita_23030630024-010.png

Pada plot2d(), terdapat beberapa gaya alternatif yang tersedia dengan grid=x. Sebagai gambaran umum, kami menampilkan berbagai gaya grid dalam satu gambar (lihat di bawah ini untuk perintah figure()). Gaya grid=0 tidak disertakan. Gaya ini tidak menampilkan grid dan frame.

>figure(3,3); . . .

> for k=1:9; figure(k); plot2d("xˆ3-x",-2,1,grid=k); end; . . .
> figure(0):



Gambar 11: images/PLOT%202D_Diva%20Nagita_23030630024-011.png

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

Atau, a, b, c, d dapat ditentukan sebagai parameter yang ditetapkan sebagai a=. . . dst.

Pada contoh berikut, kita mengubah gaya grid, menambahkan label, dan menggunakan label vertikal untuk sumbu y.

>aspect(1.5); plot2d("sin(x)",0,2pi,-1.2,1.2,grid=3,xl="x",yl="sin(x)"):

>plot2d("sin(x)+cos(2*x)",0,4pi):

Gambar yang dihasilkan dengan menyisipkan plot ke dalam jendela teks disimpan dalam direktori yang sama dengan notebook, secara default dalam subdirektori bernama "images". Gambar-gambar tersebut juga digunakan oleh ekspor HTML.

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

Fungsi atau ekspresi dalam plot2d dievaluasi secara adaptif. Untuk kecepatan yang lebih tinggi, matikan plot adaptif dengan <adaptive dan tentukan jumlah subinterval dengan n=. . . Hal ini hanya diperlukan pada kasus-kasus yang jarang terjadi.

>plot2d("sign(x)*exp(-xˆ2)",-1,1,<adaptive,n=10000):

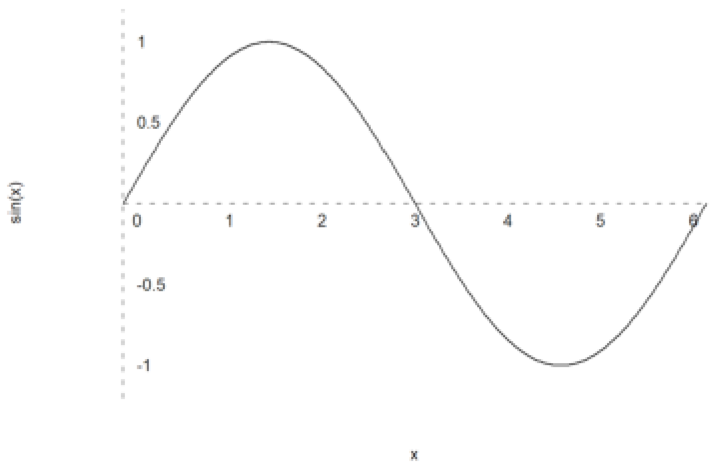
>plot2d("xˆx",r=1.2,cx=1,cy=1):

Perhatikan bahwa xˆx tidak didefinisikan untuk x<=0. Fungsi plot2d menangkap kesalahan ini, dan mulai memplot segera setelah fungsi didefinisikan. Hal ini berlaku untuk semua fungsi yang mengembalikan NAN di luar jangkauan definisinya.

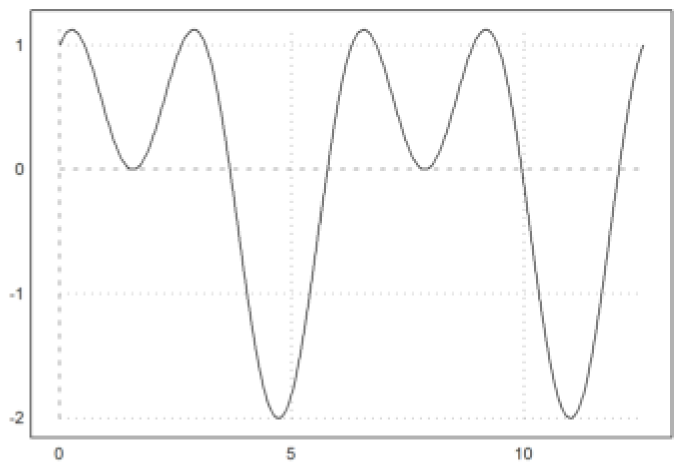
>plot2d("log(x)",-0.1,2):

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

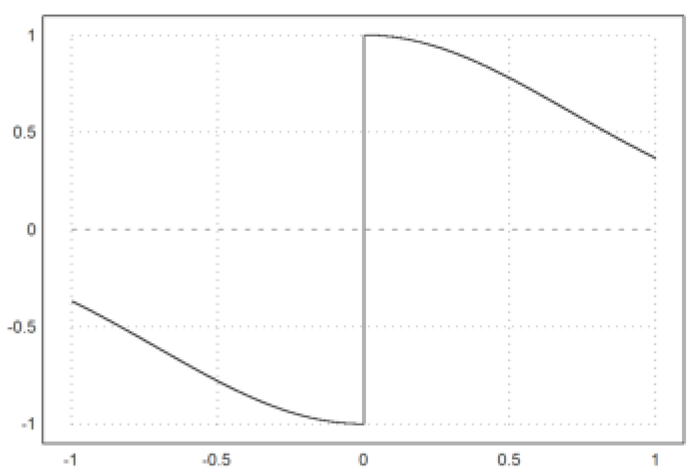
>plot2d("xˆ3-x",>square):



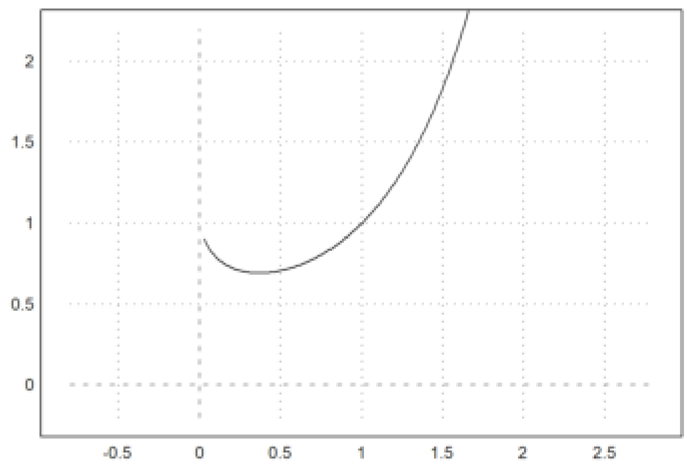
Gambar 12: images/PLOT%202D_Diva%20Nagita_23030630024-012.png



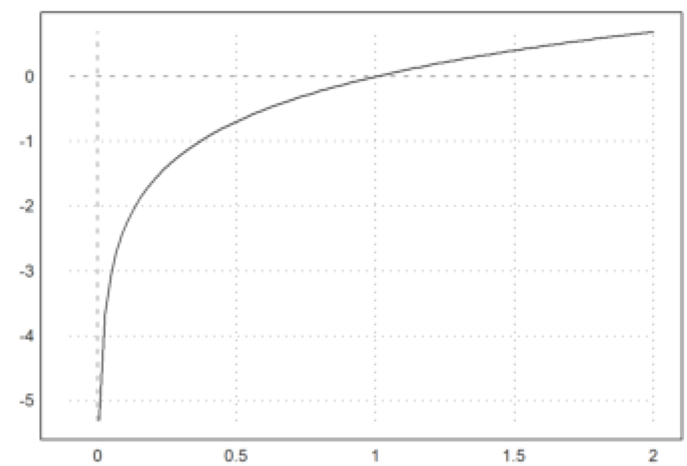
Gambar 13: images/PLOT%202D_Diva%20Nagita_23030630024-013.png



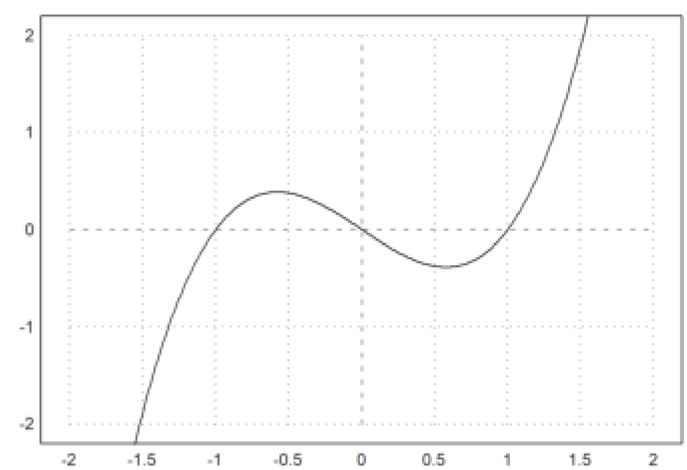
Gambar 14: images/PLOT%202D_Diva%20Nagita_23030630024-014.png



Gambar 15: images/PLOT%202D_Diva%20Nagita_23030630024-015.png

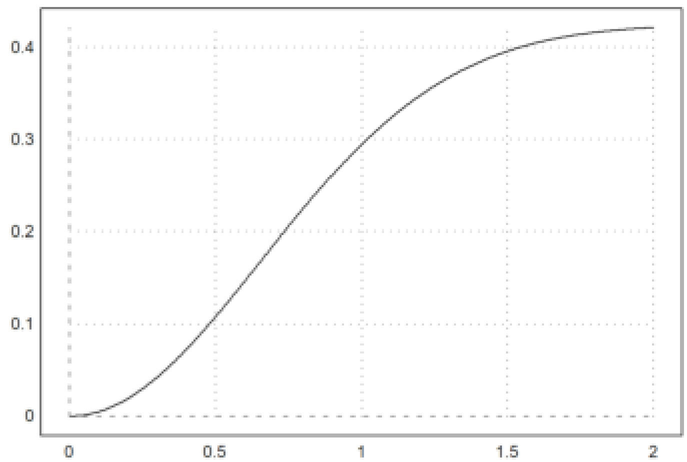


Gambar 16: images/PLOT%202D_Diva%20Nagita_23030630024-016.png



Gambar 17: images/PLOT%202D_Diva%20Nagita_23030630024-017.png

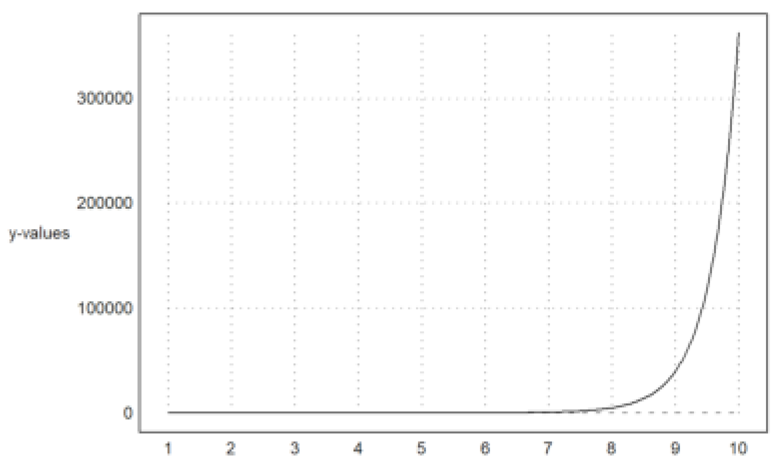
>plot2d(''integrate("sin(x)*exp(-x^2)",0,x)' ',0,2): // plot integral



Gambar 18: images/PLOT%202D_Diva%20Nagita_23030630024-018.png

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

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



Gambar 19: images/PLOT%202D_Diva%20Nagita_23030630024-019.png

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; // menentukan ekspresi

>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

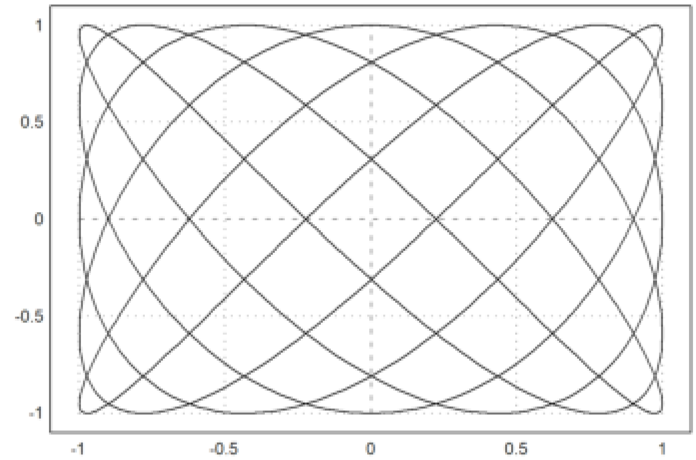
>plot2d(&diff(expr,x,2),a=-2,b=2,c=-2,d=1): // plot in rectangle

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

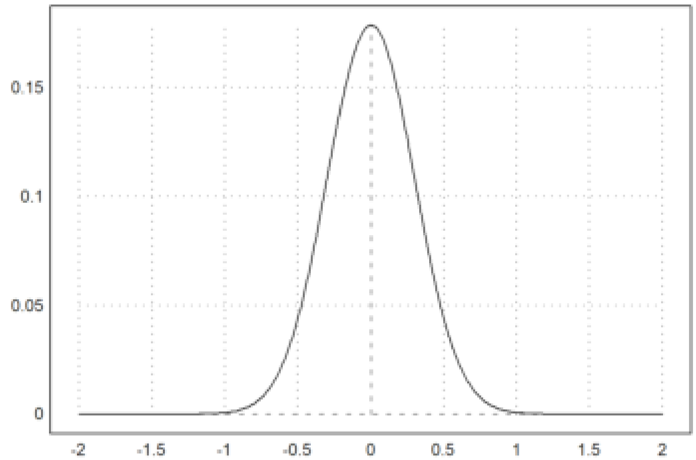
>plot2d("x^2",0,1,steps=1,color=red,n=10):

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

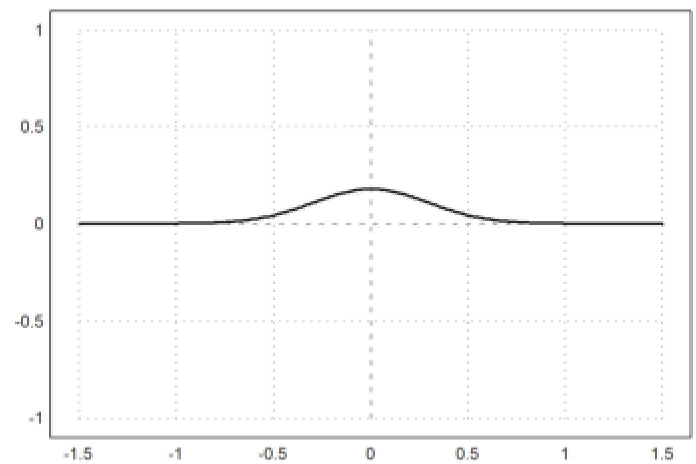
mendefinisikan ekspresi* Fungsi dalam satu Parameter



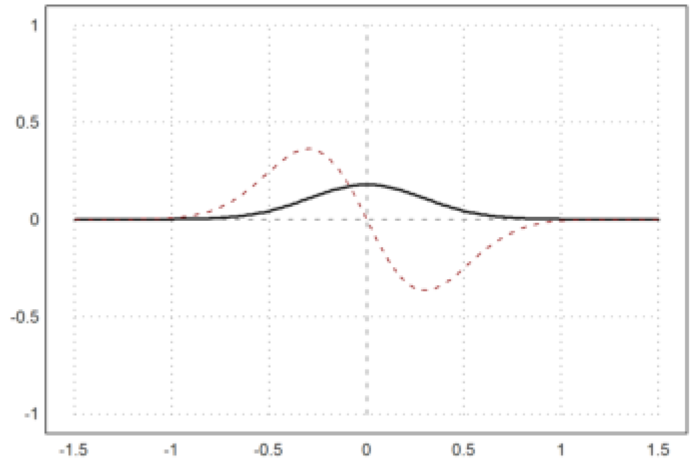
Gambar 20: images/PLOT%202D_Diva%20Nagita_23030630024-020.png



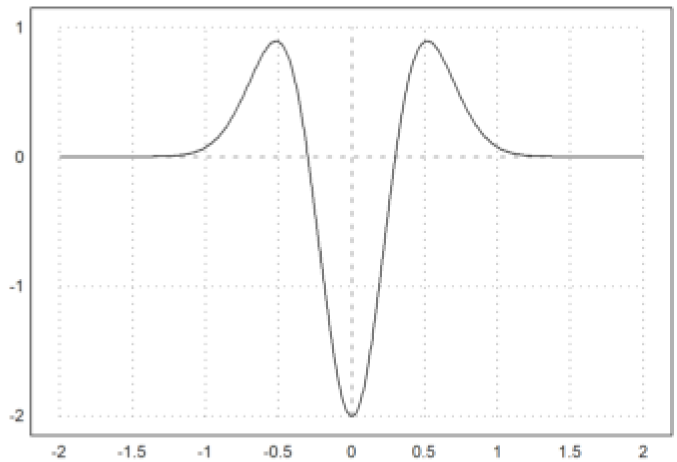
Gambar 21: images/PLOT%202D_Diva%20Nagita_23030630024-021.png



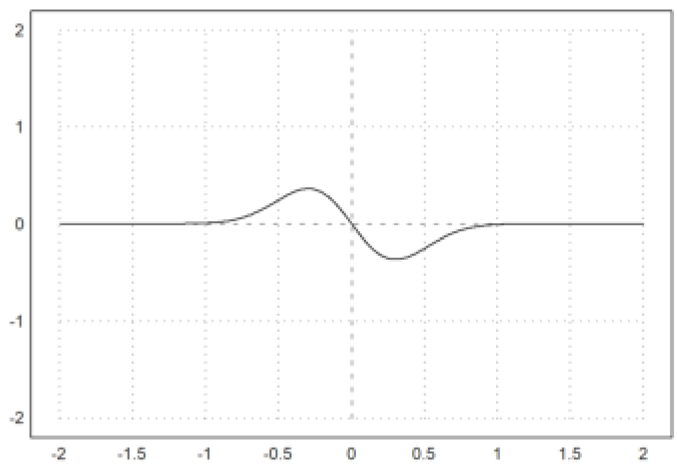
Gambar 22: images/PLOT%202D_Diva%20Nagita_23030630024-022.png



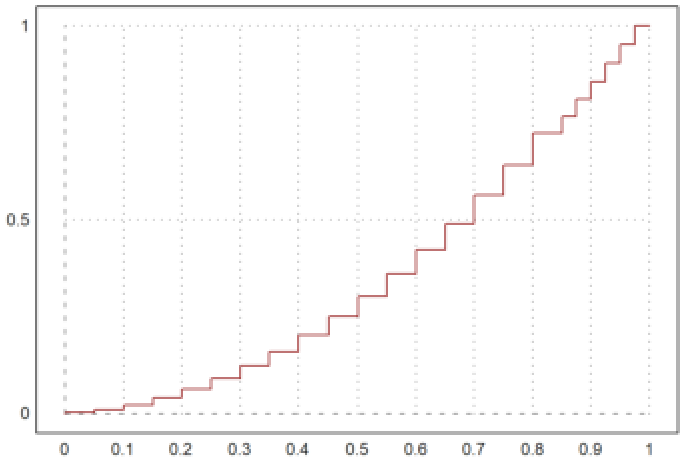
Gambar 23: images/PLOT%202D_Diva%20Nagita_23030630024-023.png



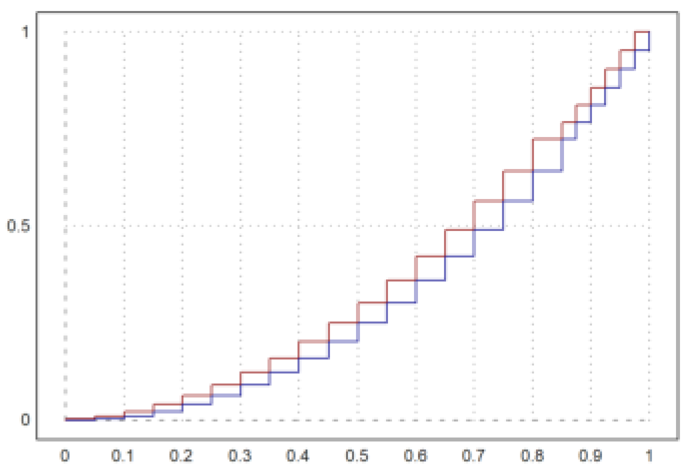
Gambar 24: images/PLOT%202D_Diva%20Nagita_23030630024-024.png



Gambar 25: images/PLOT%202D_Diva%20Nagita_23030630024-025.png



Gambar 26: images/PLOT%202D_Diva%20Nagita_23030630024-026.png



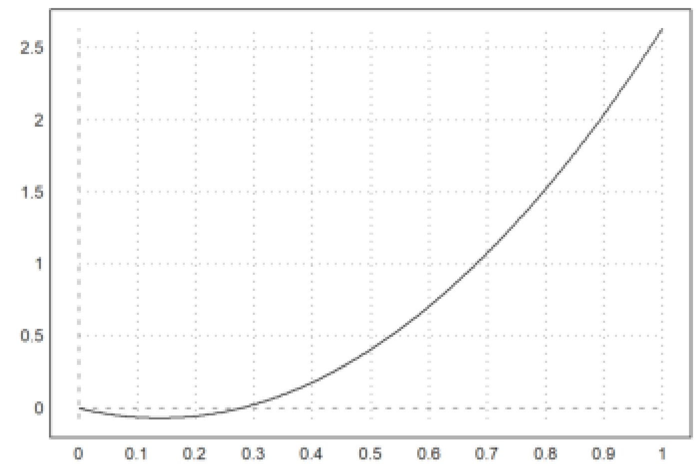
Gambar 27: images/PLOT%202D_Diva%20Nagita_23030630024-027.png

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

Berikut adalah beberapa contoh penggunaan fungsi. Seperti biasa dalam EMT, fungsi yang bekerja untuk fungsi atau ekspresi lain, Anda dapat mengoper parameter tambahan (selain x) yang bukan variabel global ke fungsi dengan parameter titik koma atau dengan koleksi panggilan.

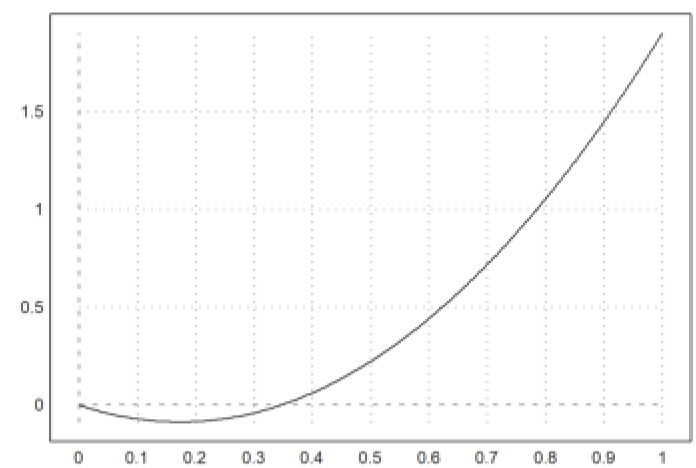
>function f(x,a) := $x^{2/a+a*x}$2-x; // define a function

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



Gambar 28: images/PLOT%202D_Diva%20Nagita_23030630024-028.png

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



Gambar 29: images/PLOT%202D_Diva%20Nagita_23030630024-029.png

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

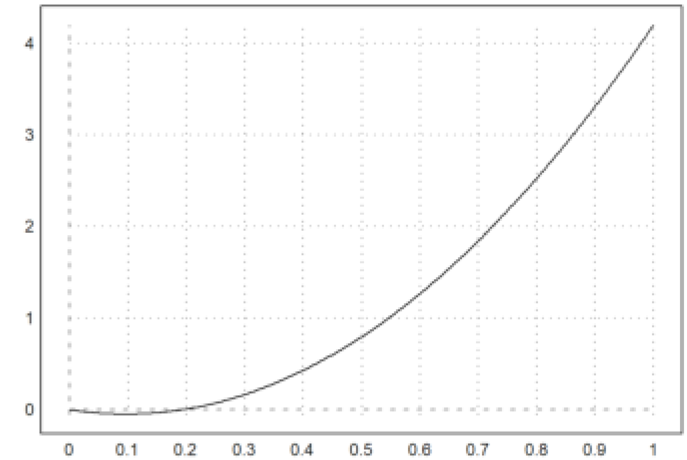
>plot2d({{“f(x,b)”,b=0.1}},0,1): // plot with 0.1

>function f(x) := xˆ3-x; . . .
> plot2d(“f”,r=1):

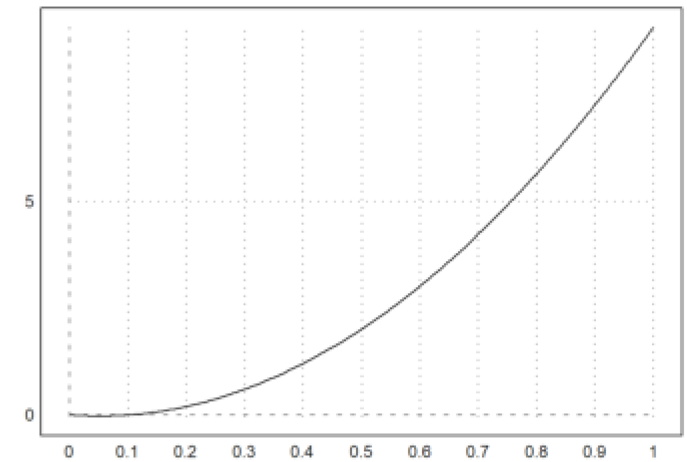
Berikut ini adalah ringkasan dari fungsi yang diterima + ekspresi atau ekspresi simbolik dalam x + fungsi atau fungsi simbolik dengan nama sebagai “f” + fungsi-fungsi simbolik hanya dengan nama f

Fungsi plot2d() juga menerima fungsi simbolik. Untuk fungsi simbolik, nama saja sudah cukup.

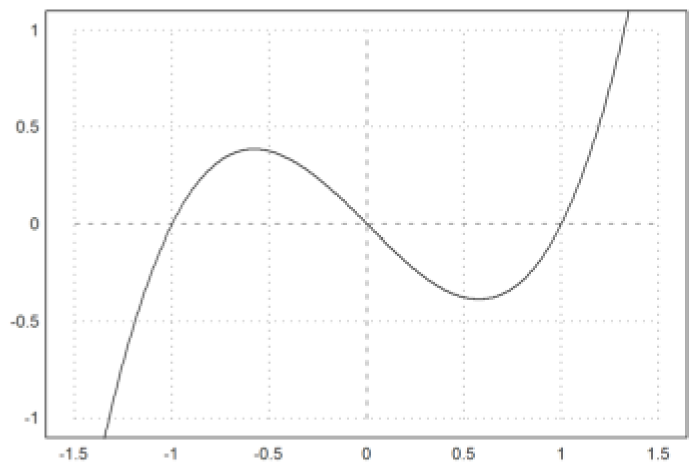
>function f(x) &= diff(xˆx,x)



Gambar 30: images/PLOT%202D_Diva%20Nagita_23030630024-030.png



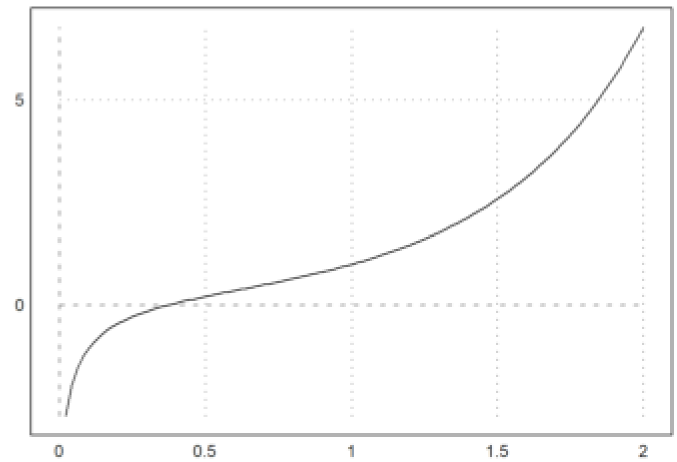
Gambar 31: images/PLOT%202D_Diva%20Nagita_23030630024-031.png



Gambar 32: images/PLOT%202D_Diva%20Nagita_23030630024-032.png

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

>plot2d(f,0,2):

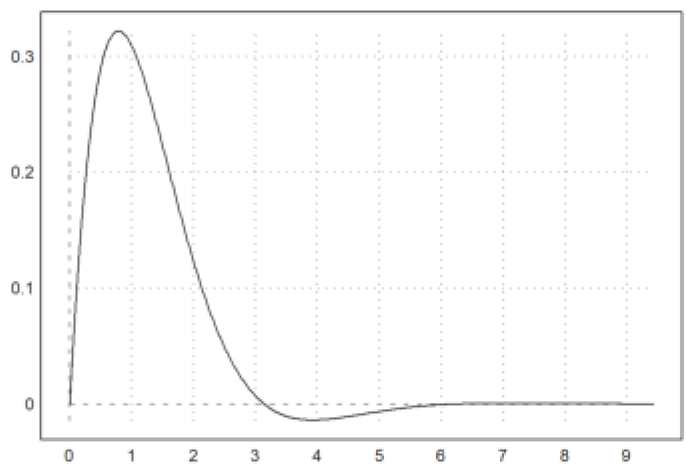


Gambar 33: images/PLOT%202D_Diva%20Nagita_23030630024-033.png

Tentu saja, untuk ekspresi atau ungkapan simbolik, nama variabel sudah cukup untuk memplotnya.
>expr &= sin(x)*exp(-x)

```
     - x
    E     sin(x)
```

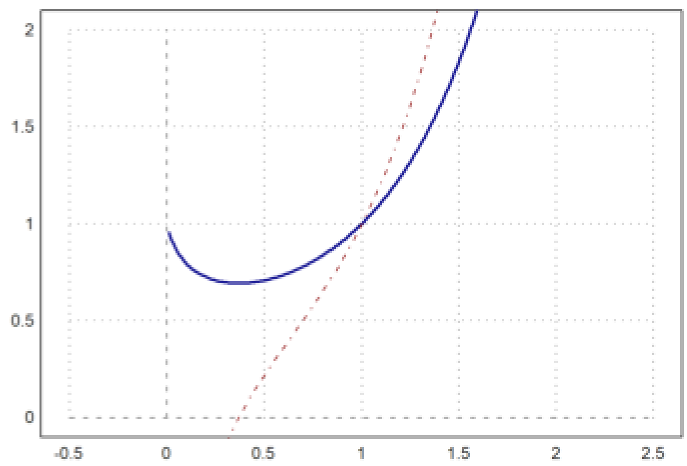
>plot2d(expr,0,3pi):



Gambar 34: images/PLOT%202D_Diva%20Nagita_23030630024-034.png

>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.

- style = "...". Pilih dari "-", "–", "-.", ".", ".-.", "-.-".



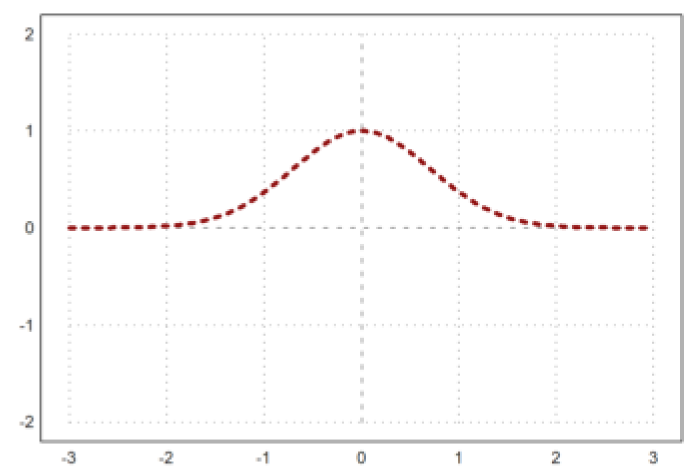
Gambar 35: images/PLOT%202D_Diva%20Nagita_23030630024-035.png

- color: Lihat di bawah untuk warna.
- ketebalan: Default 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): parameter adalah real dalam [0,1].

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



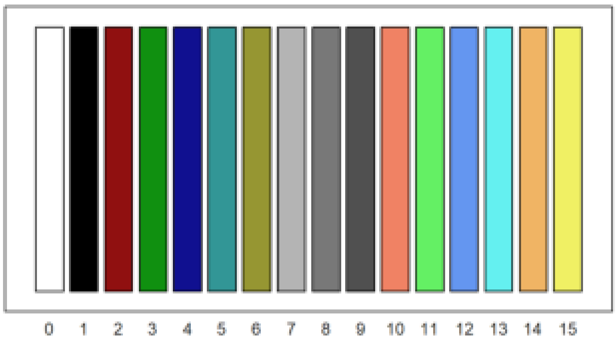
Gambar 36: images/PLOT%202D_Diva%20Nagita_23030630024-036.png

Berikut ini adalah pemandangan warna EMT yang sudah ditetapkan sebelumnya.

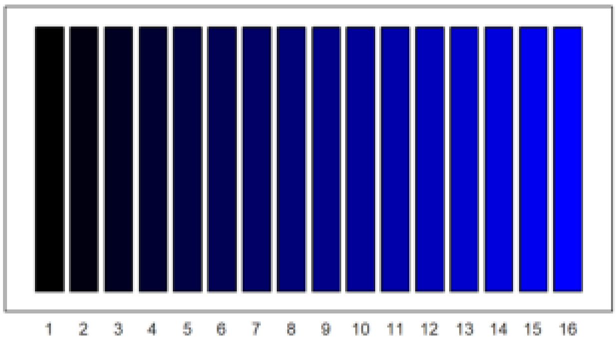
>aspect(2); columnsplot(ones(1,16),lab=0:15,grid=0,color=0:15):

Tetapi Anda bisa menggunakan warna apa pun.

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



Gambar 37: images/PLOT%202D_Diva%20Nagita_23030630024-037.png



Gambar 38: images/PLOT%202D_Diva%20Nagita_23030630024-038.png

Menggambar beberapa kurva pada bidang koordinat yang sama

Memplot lebih dari satu fungsi (beberapa fungsi) ke dalam satu jendela dapat dilakukan dengan berbagai cara. Salah satu caranya adalah dengan menggunakan >add untuk beberapa pemanggilan ke plot2d secara bersamaan, kecuali pemanggilan pertama. Kita telah menggunakan fitur ini pada contoh di atas.

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

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

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

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

Kami menambahkan titik perpotongan dengan label (pada posisi "cl" untuk kiri tengah), dan menyisipkan hasilnya ke dalam buku catatan. Kami juga menambahkan judul ke plot.

>plot2d(["cos(x)","x"],r=1.1,cx=0.5,cy=0.5, ...
> color=[black,blue],style=["-","."], ...
> grid=1);

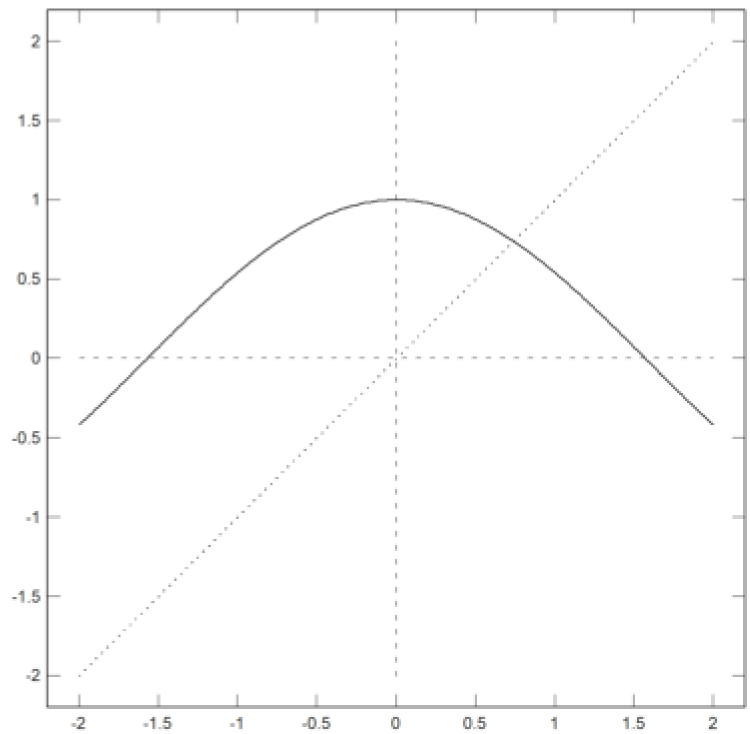
>x0=solve("cos(x)-x",1); ...
> plot2d(x0,x0,>points,>add,title="Intersection Demo"); ...
> label("cos(x) = x",x0,x0,pos="cl",offset=20):

Dalam demo berikut ini, kami memplot fungsi sinc(x)=sin(x)/x dan ekspansi Taylor ke-8 dan ke-16. Kami menghitung ekspansi ini menggunakan Maxima melalui ekspresi simbolik.

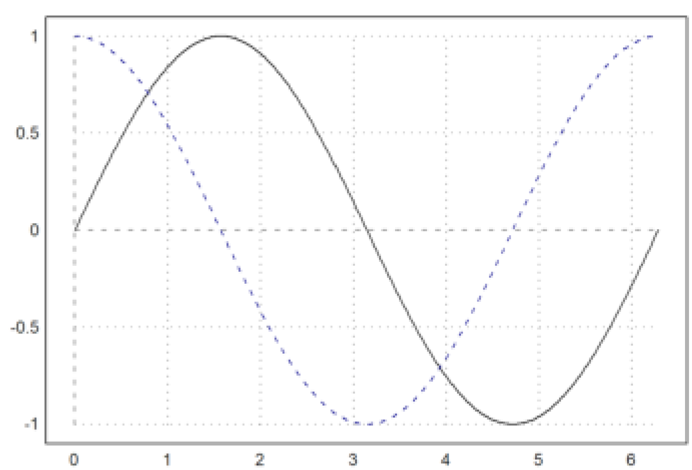
Plot ini dilakukan dalam perintah multi-baris berikut dengan tiga pemanggilan plot2d(). Perintah kedua dan ketiga memiliki set flag >add, yang membuat plot menggunakan rentang sebelumnya.

Kami menambahkan sebuah kotak label yang menjelaskan fungsi-fungsi tersebut.

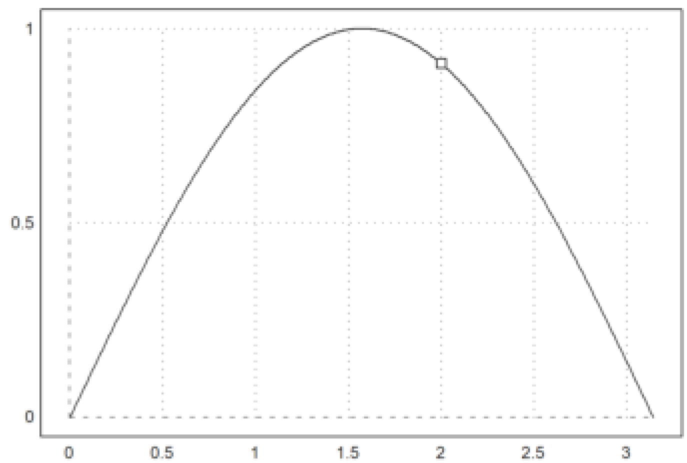
>$taylor(sin(x)/x,x,0,4)



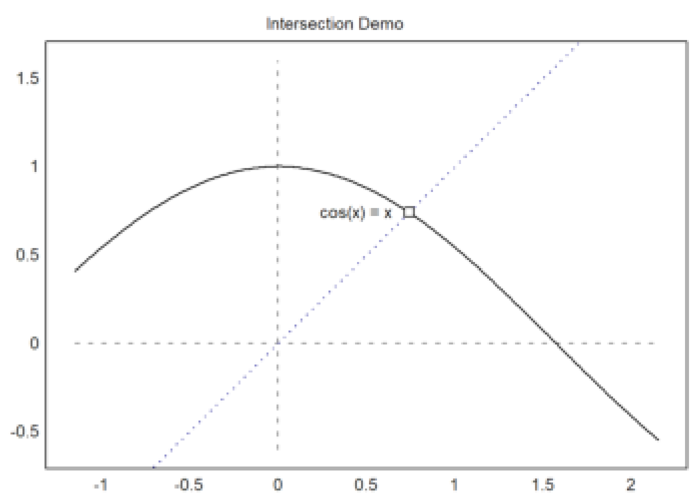
Gambar 39: images/PLOT%202D_Diva%20Nagita_23030630024-039.png



Gambar 40: images/PLOT%202D_Diva%20Nagita_23030630024-040.png



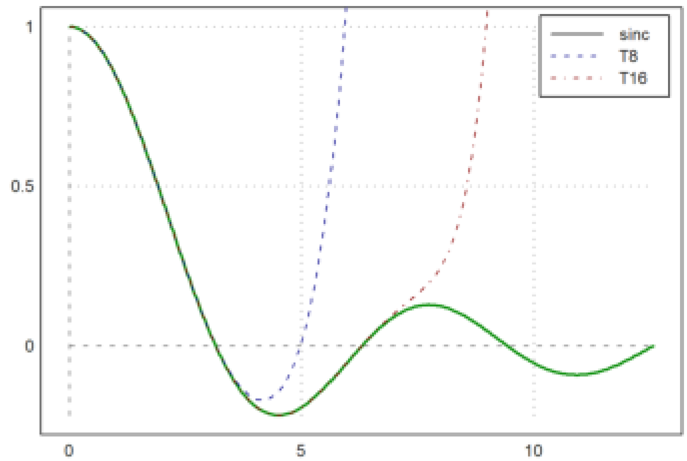
Gambar 41: images/PLOT%202D_Diva%20Nagita_23030630024-041.png



Gambar 42: images/PLOT%202D_Diva%20Nagita_23030630024-042.png

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

>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]):

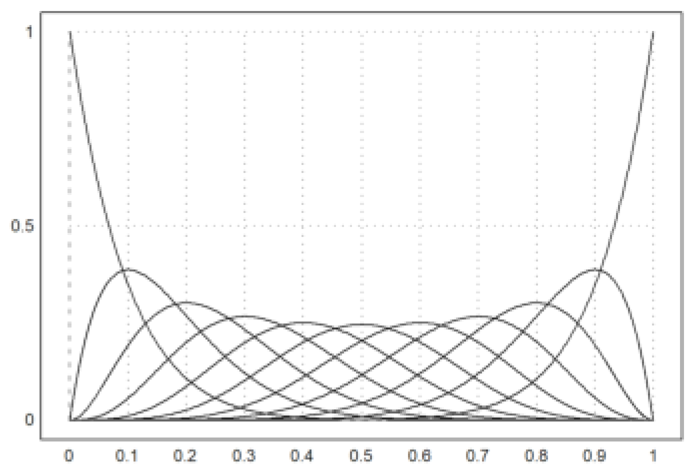


Gambar 43: images/PLOT%202D_Diva%20Nagita_23030630024-044.png

Pada contoh berikut, kami menghasilkan Polinomial Bernstein.

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

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



Gambar 44: images/PLOT%202D_Diva%20Nagita_23030630024-046.png

Metode kedua menggunakan sepasang matriks nilai x dan matriks nilai y dengan ukuran yang sama.

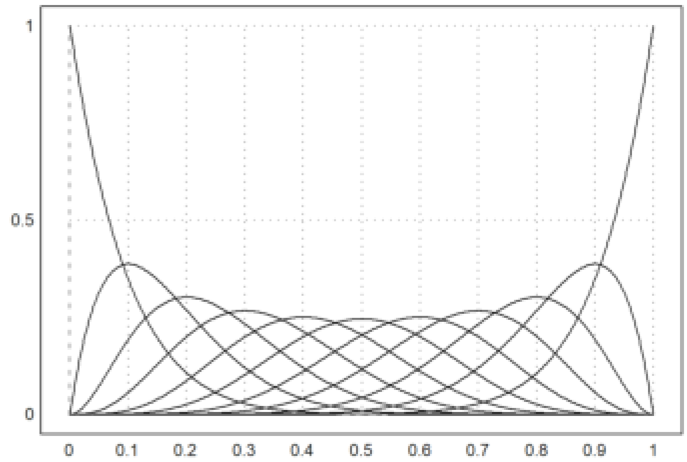
Kita membuat sebuah matriks nilai dengan satu Polinomial Bernstein di setiap baris. Untuk ini, kita cukup menggunakan vektor kolom i. Lihatlah pengantar tentang bahasa matriks untuk mempelajari lebih lanjut.

>x=linspace(0,1,500);

>n=10; k=(0:n)'; // n is row vector, k is column vector

>y=bin(n,k)*x$^{k*(1-x)}$(n-k); // y is a matrix then

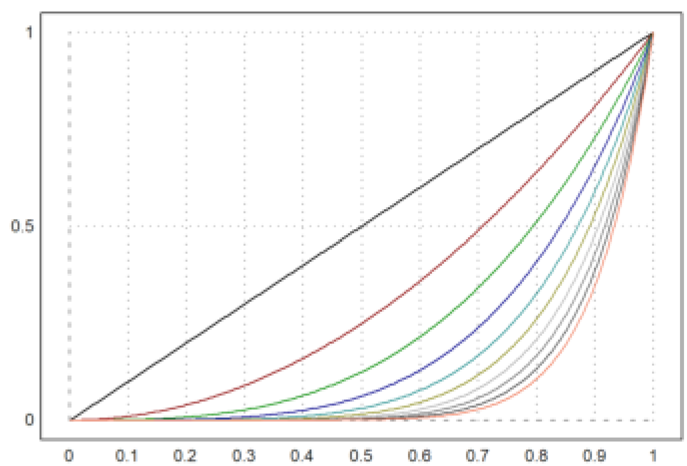
>plot2d(x,y):



Gambar 45: images/PLOT%202D_Diva%20Nagita_23030630024-047.png

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):



Gambar 46: images/PLOT%202D_Diva%20Nagita_23030630024-048.png

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

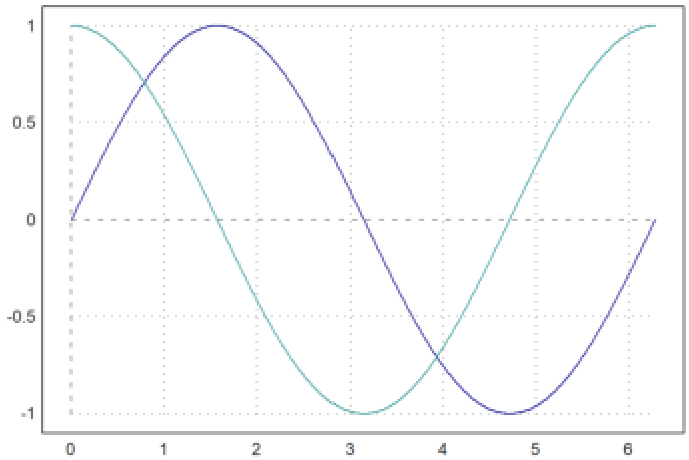
>plot2d(["sin(x)","cos(x)"],0,2pi,color=4:5):

>plot2d(["sin(x)","cos(x)"],0,2pi): // plot vector of expressions

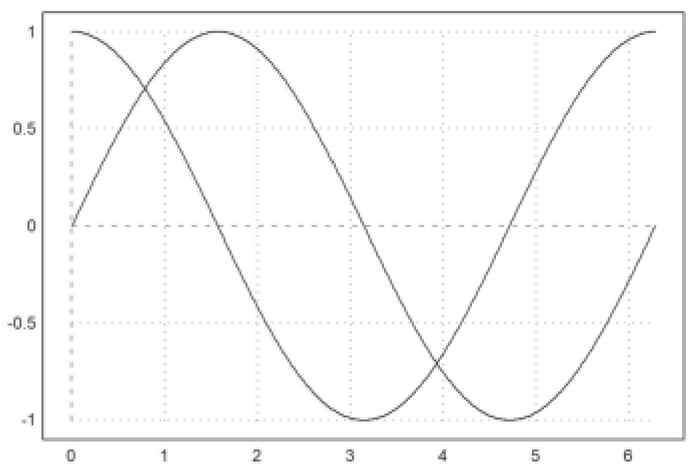
Kita bisa mendapatkan vektor seperti itu dari Maxima dengan menggunakan makelist() dan mxm2str().

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

```
            10              9                 8   2                 7   3
[(1 - x)    , 10 (1 - x)   x, 45 (1 - x)   x , 120 (1 - x)   x ,
```



Gambar 47: images/PLOT%202D_Diva%20Nagita_23030630024-049.png



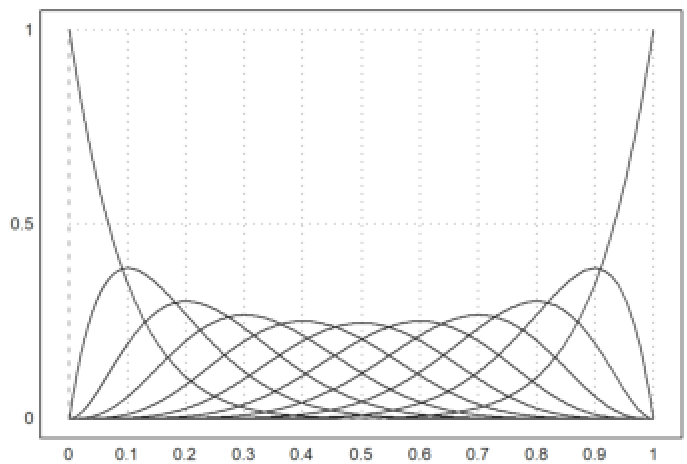
Gambar 48: images/PLOT%202D_Diva%20Nagita_23030630024-050.png

```
           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



Gambar 49: images/PLOT%202D_Diva%20Nagita_23030630024-051.png

Alternatif lain adalah dengan menggunakan bahasa matriks Euler.

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

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

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

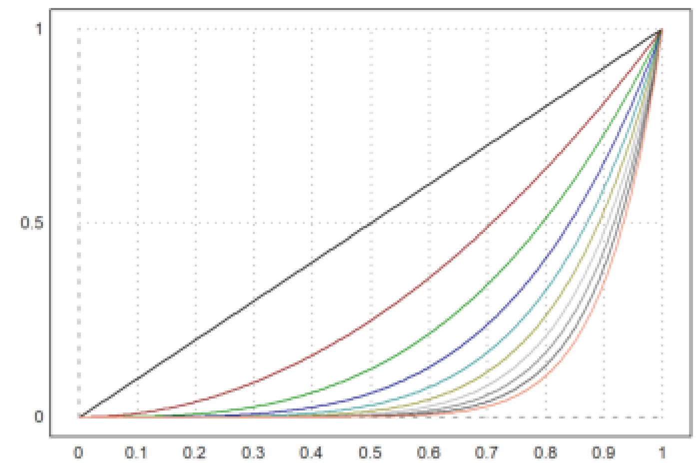
Ekspresi dan fungsi satu baris dapat melihat variabel global.

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

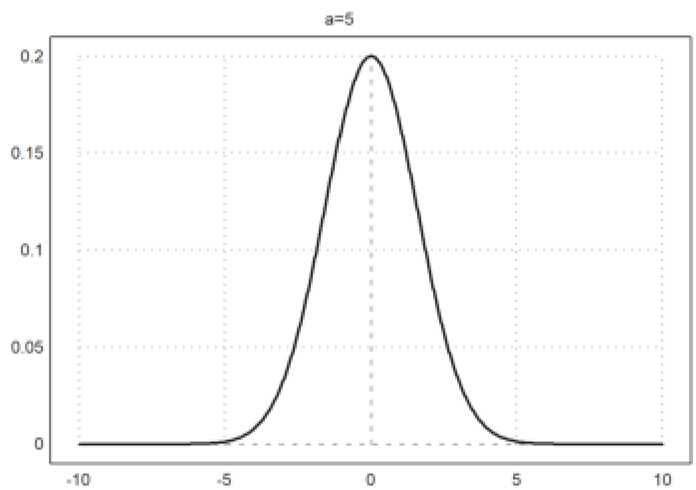
Berhati-hatilah untuk meletakkan semua parameter yang diberikan di akhir perintah plot2d. Pada contoh ini kita mengoper a=5 ke fungsi f, yang kita plot dari -10 ke 10.

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

Atau, gunakan koleksi dengan nama fungsi dan semua parameter tambahan. Daftar khusus ini disebut koleksi panggilan, dan itu adalah cara yang lebih disukai untuk mengoper argumen ke fungsi yang dengan sendirinya dioper sebagai argumen ke fungsi lain.



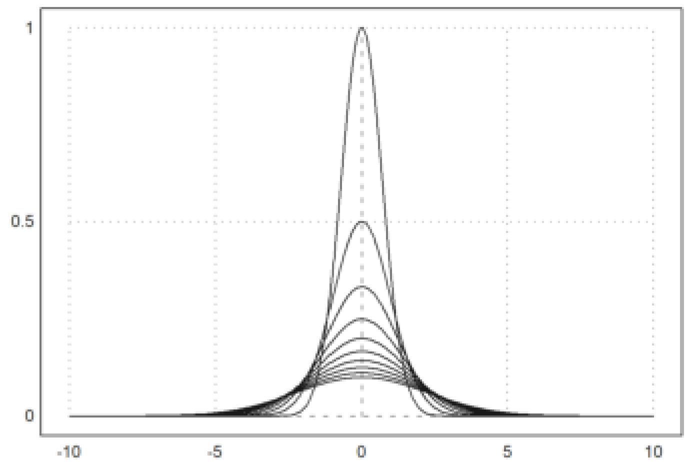
Gambar 50: images/PLOT%202D_Diva%20Nagita_23030630024-052.png



Gambar 51: images/PLOT%202D_Diva%20Nagita_23030630024-053.png

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

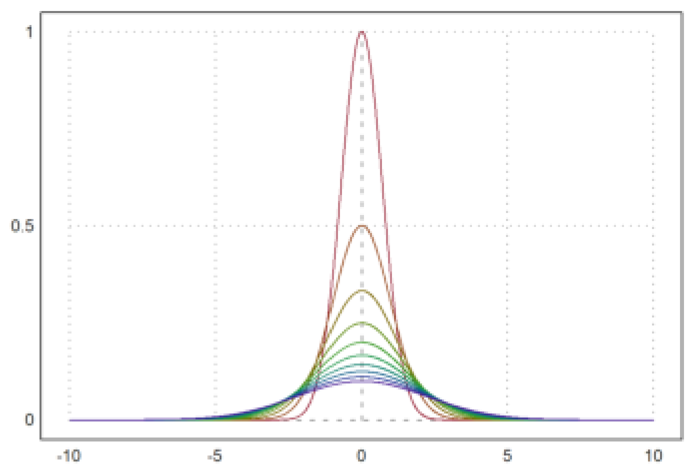
>plot2d({{"f",1}},-10,10); . . .
> for a=2:10; plot2d({{"f",a}},>add); end:



Gambar 52: images/PLOT%202D_Diva%20Nagita_23030630024-054.png

Kita dapat mencapai hasil yang sama dengan cara berikut menggunakan bahasa matriks EMT. Setiap baris dari matriks f(x,a) adalah satu fungsi. Selain itu, kita dapat mengatur warna untuk setiap baris matriks. Klik dua kali pada fungsi getspectral() untuk penjelasannya.

>x=-10:0.01:10; a=(1:10)'; plot2d(x,f(x,a),color=getspectral(a/10)):



Gambar 53: images/PLOT%202D_Diva%20Nagita_23030630024-055.png

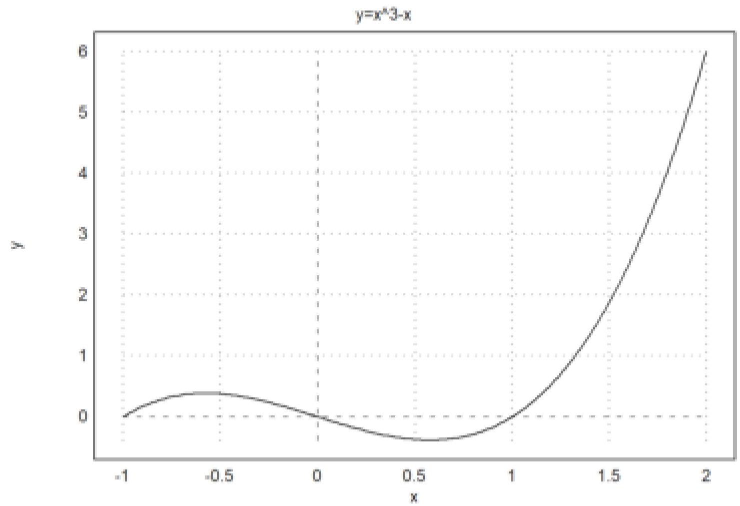
Label Teks

Dekorasi sederhana dapat berupa

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

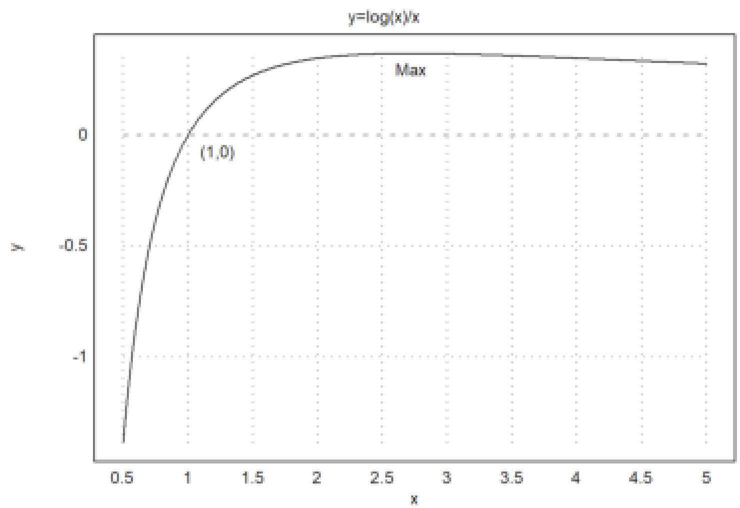
Perintah label akan memplotkan ke dalam plot saat ini pada koordinat plot (x,y). Perintah ini dapat menerima sebuah argumen posisi.

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



Gambar 54: images/PLOT%202D_Diva%20Nagita_23030630024-056.png

>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"):



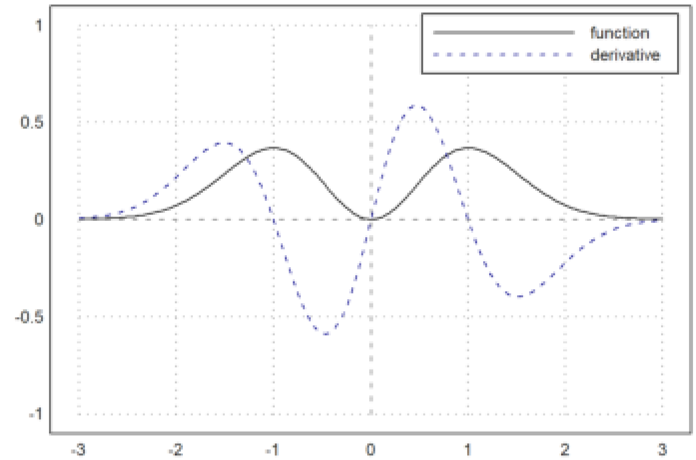
Gambar 55: images/PLOT%202D_Diva%20Nagita_23030630024-057.png

Ada juga fungsi labelbox(), yang dapat menampilkan fungsi dan teks. Fungsi ini membutuhkan vektor string dan warna, satu item untuk setiap fungsi.

>function f(x) &= x$^{2*exp(-x}$2); ...
> plot2d(&f(x),a=-3,b=3,c=-1,d=1); ...
> plot2d(&diff(f(x),x),>add,color=blue,style="–"); ...
> labelbox(["function","derivative"],styles=["-","–"], ...
> colors=[black,blue],w=0.4):

Kotak tersebut berlabuh di kanan atas secara default, tetapi >kiri menambatkannya di kiri atas. Anda dapat memindahkannya ke tempat mana pun yang Anda suka. Posisi jangkar adalah sudut kanan atas kotak, dan angkanya adalah pecahan dari ukuran jendela grafik. Lebarnya otomatis.

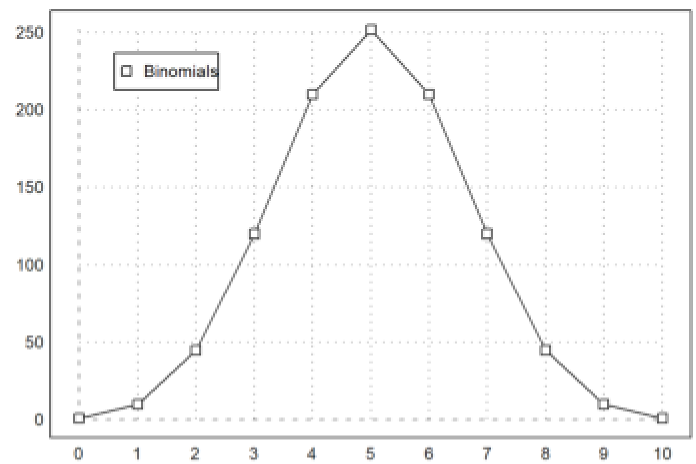
Untuk plot titik, kotak label juga dapat digunakan. Tambahkan parameter >titik, atau vektor bendera, satu untuk setiap label.



Gambar 56: images/PLOT%202D_Diva%20Nagita_23030630024-058.png

Pada contoh berikut, hanya ada satu fungsi. Jadi kita dapat menggunakan string sebagai pengganti vektor string. Kita 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):



Gambar 57: images/PLOT%202D_Diva%20Nagita_23030630024-059.png

Gaya plot ini juga tersedia di statplot(). Seperti pada plot2d() warna dapat diatur untuk setiap baris plot. Terdapat lebih banyak plot khusus untuk keperluan statistik (lihat tutorial tentang statistik).

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

Fitur yang serupa adalah fungsi textbox().

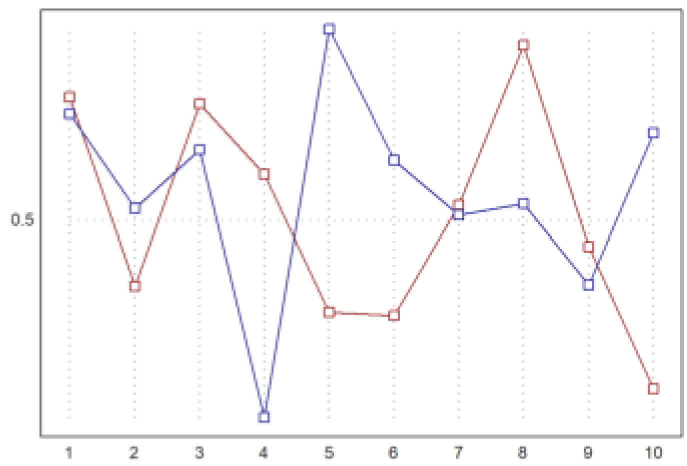
Lebarnya secara default adalah lebar maksimal baris teks. Tetapi bisa juga diatur oleh pengguna.

>function f(x) &= exp(-x)*sin(2*pi*x); . . .
> plot2d("f(x)",0,2pi); . . .
> textbox(latex("\text{Example of a damped oscillation}\ f(x)=e^{-x}sin(2\pi x)"),w=0.85):

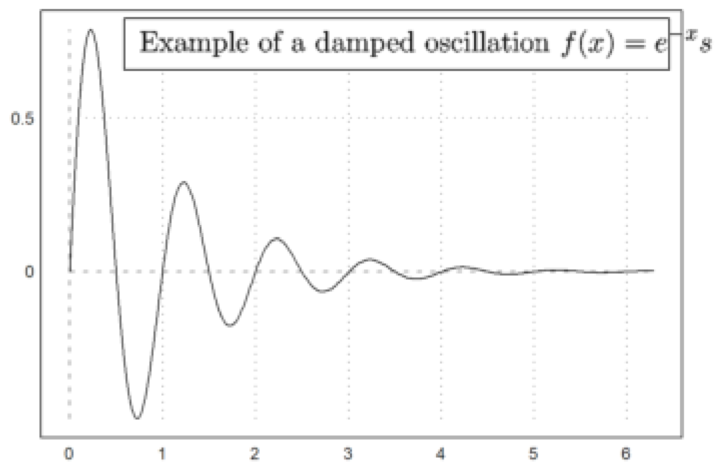
Label teks, judul, kotak label, dan teks lainnya dapat berisi string Unicode (lihat sintaks EMT untuk mengetahui lebih lanjut tentang string Unicode).

>plot2d("x^3-x",title=u"x → x³ - x"):

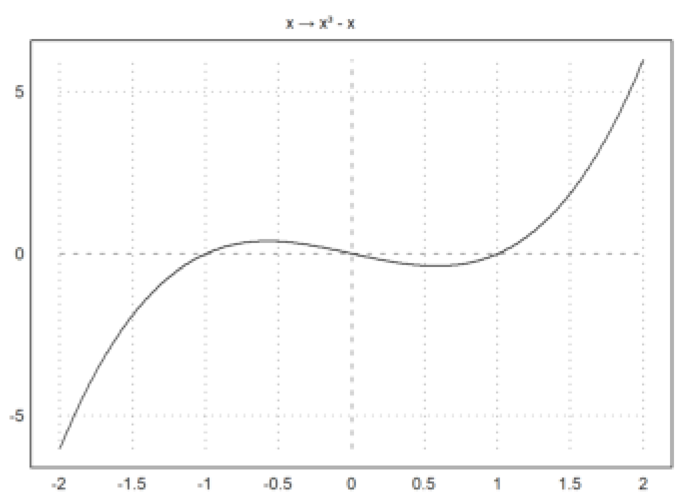
Label pada sumbu x dan y bisa vertikal, begitu juga dengan sumbu.



Gambar 58: images/PLOT%202D_Diva%20Nagita_23030630024-060.png

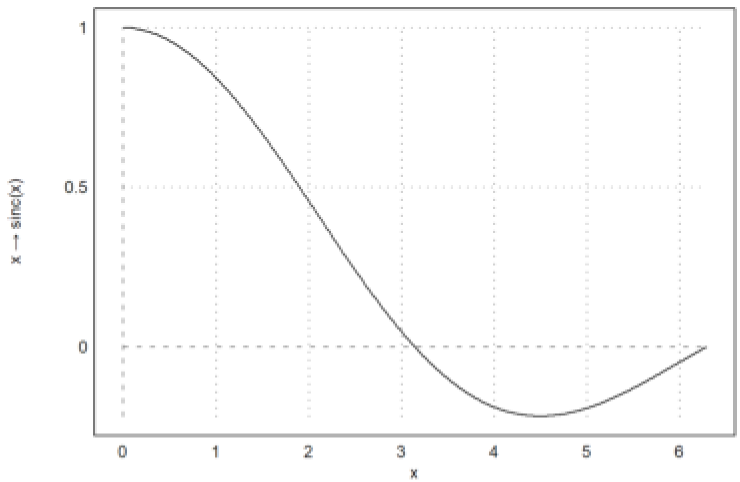


Gambar 59: images/PLOT%202D_Diva%20Nagita_23030630024-061.png



Gambar 60: images/PLOT%202D_Diva%20Nagita_23030630024-062.png

>plot2d("sinc(x)",0,2pi,xl="x",yl=u"x → sinc(x)",>vertical):



Gambar 61: images/PLOT%202D_Diva%20Nagita_23030630024-063.png

LaTeX

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

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

Pada plot berikut ini, kita menggunakan LaTeX untuk label x dan y, sebuah 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 sesuai pada sumbu x. Kita dapat menggunakan xaxis() dan yaxis() seperti yang akan kita tunjukkan nanti.

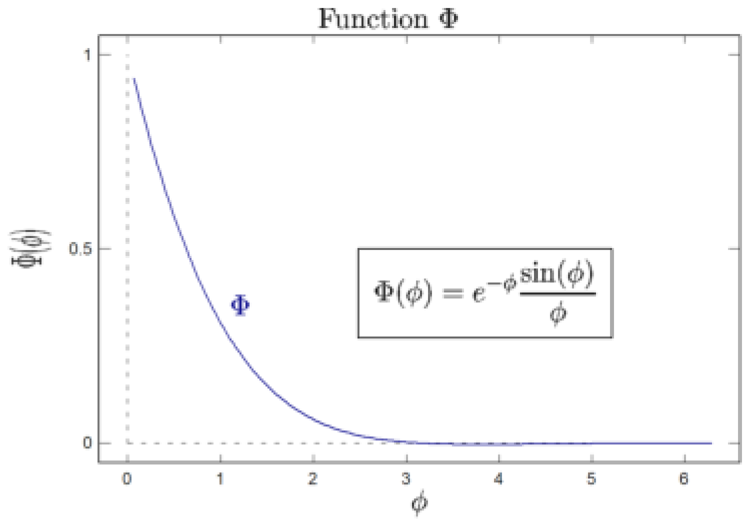
Cara termudah adalah dengan membuat plot kosong dengan sebuah frame menggunakan grid=4, dan kemudian menambahkan grid dengan ygrid() dan xgrid(). Pada contoh berikut, kita menggunakan tiga buah string LaTeX untuk label pada sumbu x dengan xtick().

>plot2d("sinc(x)",0,2pi,grid=4,<ticks); ...
> ygrid(-2:0.5:2,grid=6); ...
> xgrid([0:2]*pi,<ticks,grid=6); ...
> xtick([0,pi,2pi],["0","\pi","2\pi"],>latex):

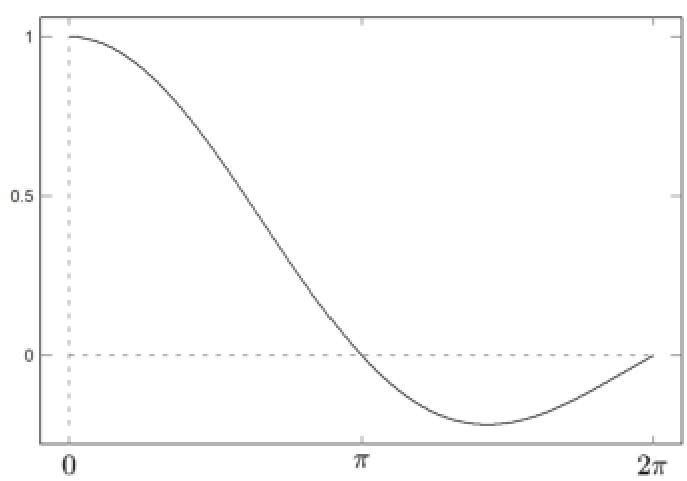
Tentu saja, fungsi juga dapat digunakan.

>function map f(x) ...

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



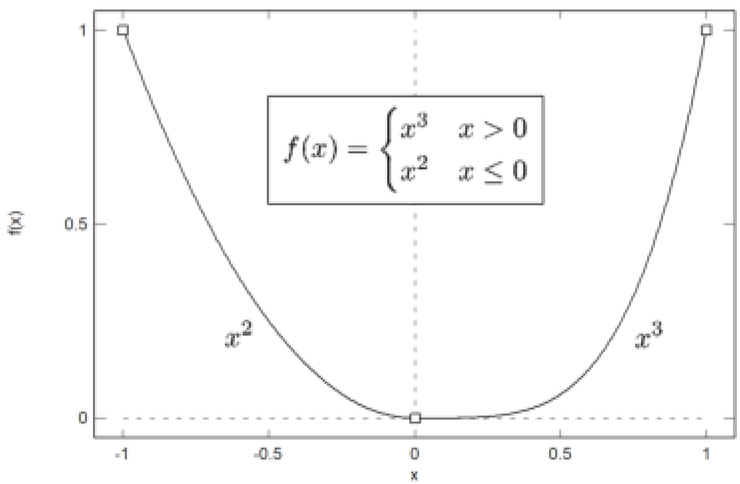
Gambar 62: images/PLOT%202D_Diva%20Nagita_23030630024-064.png



Gambar 63: images/PLOT%202D_Diva%20Nagita_23030630024-065.png

Parameter "map" membantu menggunakan fungsi untuk vektor. Untuk
plot, hal ini tidak diperlukan. Tetapi untuk mendemonstrasikan bahwa vektorisasi
berguna, kami menambahkan beberapa titik kunci pada plot pada x=-1, x=0 dan x=1.
Pada plot berikut, kita juga memasukkan beberapa kode LaTeX. Kita menggunakannya untuk
dua label dan sebuah kotak teks. Tentu saja, Anda hanya dapat menggunakan
LaTeX jika Anda telah menginstal LaTeX dengan benar.

```
>plot2d("f",-1,1,xl="x",yl="f(x)",grid=6); ...
> plot2d([-1,0,1],f([-1,0,1]),>points,>add); ...
> label(latex("x^3"),0.72,f(0.72)); ...
> label(latex("x^2"),-0.52,f(-0.52),pos="ll"); ...
> textbox( ...
> latex("f(x)=\begin{cases} x^3 & x>0 \\ x^2 & x \le 0\end{cases}"), ...
> x=0.7,y=0.2):
```



Gambar 64: images/PLOT%202D_Diva%20Nagita_23030630024-066.png

Interaksi Pengguna

Ketika memplot fungsi atau ekspresi, parameter >user memungkinkan pengguna untuk memperbesar dan menggeser plot dengan tombol kursor atau mouse. Pengguna dapat + memperbesar dengan + atau - + memindahkan plot dengan tombol kursor + memilih jendela plot dengan mouse + mengatur ulang tampilan dengan spasi + keluar dengan return

Tombol spasi akan mengatur ulang plot ke jendela plot awal.

Ketika memplot data, bendera >user hanya akan menunggu penekanan tombol.

Translated with DeepL.com (free version)

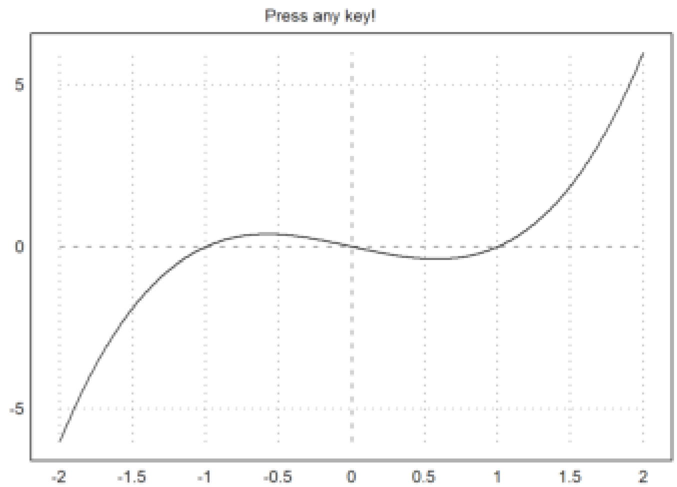
```
>plot2d({{"x^3-a*x",a=1}},>user,title="Press any key!"):
```

```
>plot2d("exp(x)*sin(x)",user=true, ...
> title="+/- or cursor keys (return to exit)"):
```

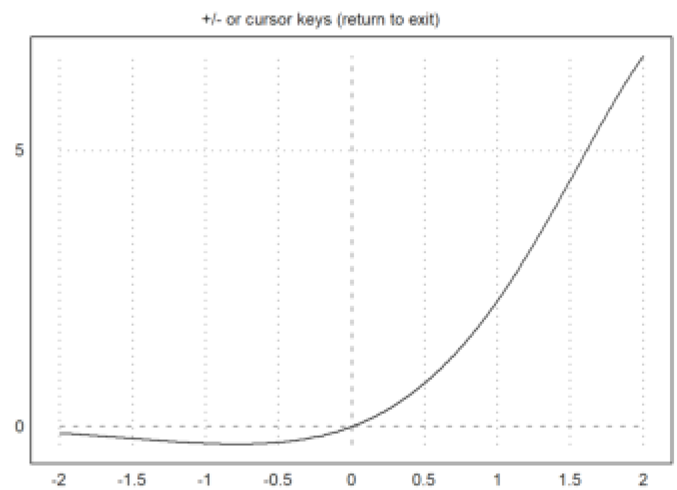
Berikut ini menunjukkan cara interaksi pengguna tingkat lanjut (lihat tutorial tentang pemrograman untuk detailnya).

Fungsi bawaan mousedrag() menunggu peristiwa mouse atau keyboard. Fungsi ini melaporkan mouse ke bawah, mouse bergerak atau mouse ke atas, dan penekanan tombol. Fungsi dragpoints() memanfaatkan hal ini, dan mengizinkan pengguna untuk menyeret titik manapun di dalam plot.

Kita membutuhkan fungsi plot terlebih dahulu. Sebagai contoh, kita melakukan interpolasi pada 5 titik dengan sebuah polinomial. Fungsi ini harus memplot ke dalam area plot yang tetap. Parameter "map" mem-



Gambar 65: images/PLOT%202D_Diva%20Nagita_23030630024-067.png



Gambar 66: images/PLOT%202D_Diva%20Nagita_23030630024-068.png

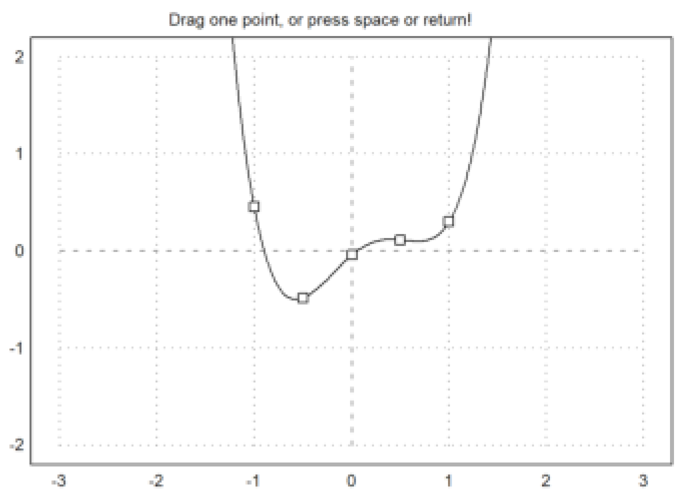
bantu untuk menggunakan fungsi ini untuk vektor. Untuk
plot, hal ini tidak diperlukan. Tetapi untuk mendemonstrasikan bahwa vektorisasi
berguna, kami menambahkan beberapa titik kunci pada plot pada x=-1, x=0 dan x=1.
Pada plot berikut, kita juga memasukkan beberapa kode LaTeX. Kita menggunakannya untuk
dua label dan sebuah kotak teks. Tentu saja, Anda hanya dapat menggunakan
LaTeX jika Anda telah menginstal LaTeX dengan benar.
>function plotf(xp,yp,select) . . .

```
  d=interp(xp,yp);
  plot2d("interpval(xp,d,x)";d,xp,r=2);
  plot2d(xp,yp,>points,>add);
  if select>0 then
    plot2d(xp[select],yp[select],color=red,>points,>add);
  endif;
  title("Drag one point, or press space or return!");
endfunction
```

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

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

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



Gambar 67: images/PLOT%202D_Diva%20Nagita_23030630024-069.png

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

Pertama, kita memerlukan fungsi plot.

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

Kemudian kita membutuhkan nama untuk parameter, nilai awal dan matriks rentang nx2, dan secara opsional, sebuah garis judul.

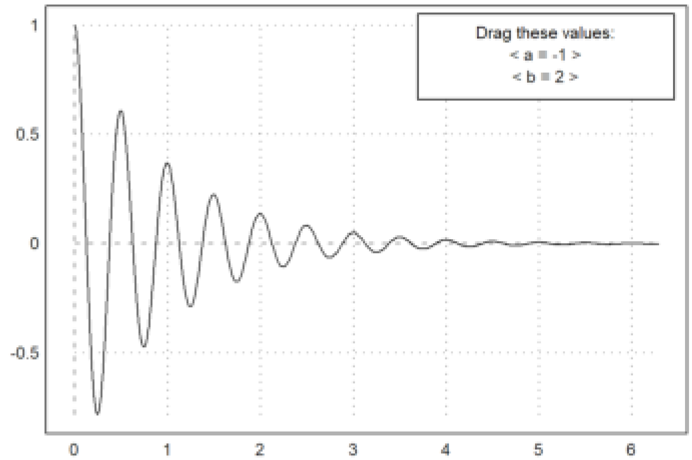
Terdapat slider interaktif, yang dapat mengatur nilai oleh pengguna. Fungsi dragvalues() menyediakan ini.

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

Anda dapat membatasi nilai yang diseret menjadi bilangan bulat. Sebagai contoh, kita menulis fungsi plot, yang memplot polinomial Taylor dengan derajat n ke fungsi kosinus.

>function plotf(n) . . .

```
plot2d("cos(x)",0,2pi,>square,grid=6);
```

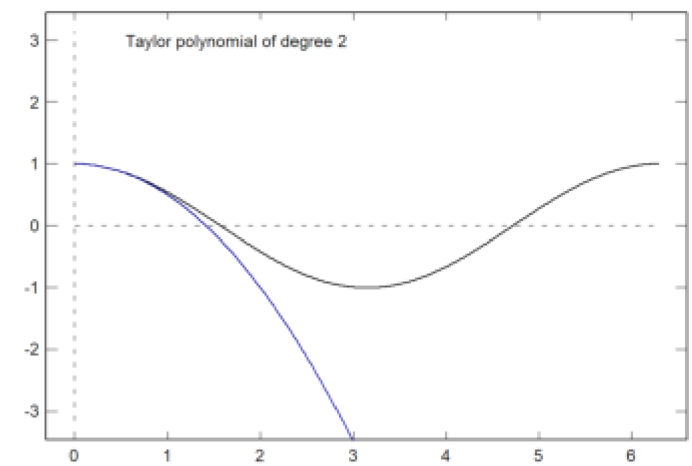


Gambar 68: images/PLOT%202D_Diva%20Nagita_23030630024-070.png

```
plot2d(&"taylor(cos(x),x,0,@n)",color=blue,>add);
textbox("Taylor polynomial of degree "+n,0.1,0.02,style="t",>left);
endfunction
```

Sekarang kita membiarkan derajat n bervariasi dari 0 sampai 20 dalam 20 stop. Hasil dari dragvalues() digunakan untuk memplot sketsa dengan n ini, dan untuk menyisipkan plot ke dalam buku catatan.

>nd=dragvalues("plotf","degree",2,[0,20],20,y=0.8, . . .
> heading="Drag the value:"); . . .
> plotf(nd):



Gambar 69: images/PLOT%202D_Diva%20Nagita_23030630024-071.png

Berikut ini adalah peragaan sederhana dari fungsi ini. Pengguna dapat menggambar di atas jendela plot, meninggalkan jejak titik.

>function dragtest . . .

```
  plot2d(none,r=1,title="Drag with the mouse, or press any key!");
  start=0;
  repeat
    {flag,m,time}=mousedrag();
    if flag==0 then return; endif;
```

```
    if flag==2 then
      hold on; mark(m[1],m[2]); hold off;
    endif;
  end
endfunction
```

>dragtest // lihat hasilnya dan cobalah lakukan!

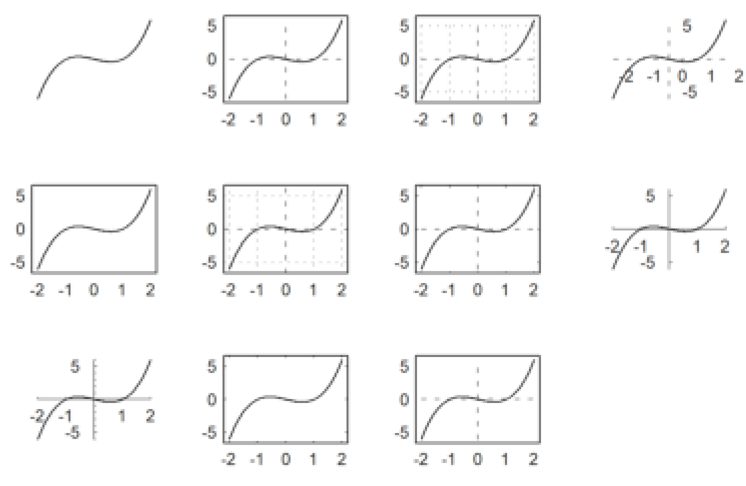
Gaya Plot 2D

Secara default, EMT menghitung tanda sumbu otomatis dan menambahkan label pada setiap tanda. Hal ini dapat diubah dengan parameter grid. Gaya default sumbu dan label dapat dimodifikasi. Selain itu, label dan judul dapat ditambahkan secara manual. Untuk mengatur ulang ke gaya default, gunakan reset().

>caspect();

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

>figure(3,4); . . .
> figure(1); plot2d("x^3-x",grid=0); . . . // tidak ada grid, bingkai atau sumbu
> figure(2); plot2d("x^3-x",grid=1); . . . // sumbu x-y
> figure(3); plot2d("x^3-x",grid=2); . . . // centang default
> figure(4); plot2d("x^3-x",grid=3); . . . // sumbu x-y dengan label di dalamnya
> figure(5); plot2d("x^3-x",grid=4); . . . // tidak ada tanda centang, hanya label
> figure(6); plot2d("x^3-x",grid=5); . . . // default, tetapi tidak ada margin
> figure(7); plot2d("x^3-x",grid=6); . . . // sumbu saja
> figure(8); plot2d("x^3-x",grid=7); . . . // sumbu saja, centang pada sumbu
> figure(9); plot2d("x^3-x",grid=8); . . . // sumbu saja, centang halus pada sumbu
> figure(10); plot2d("x^3-x",grid=9); . . . // default, centang kecil di dalam
> figure(11); plot2d("x^3-x",grid=10); . . . // tidak ada tanda centang, hanya sumbu
> figure(0):



Gambar 70: images/PLOT%202D_Diva%20Nagita_23030630024-072.png

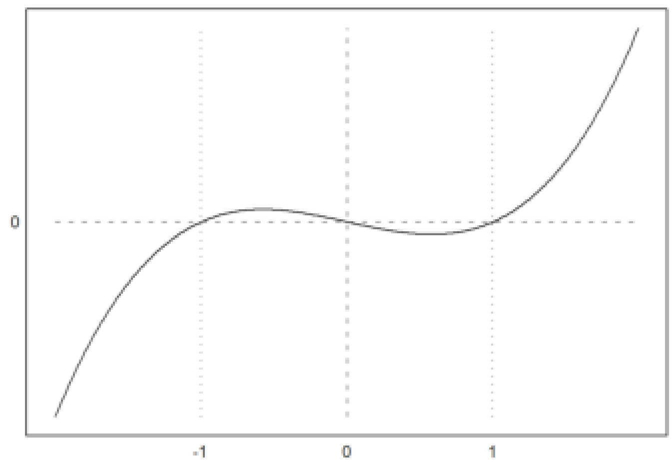
Parameter <frame mematikan bingkai, dan framecolor=blue menetapkan bingkai ke warna biru.

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

>aspect(1.5);

>plot2d("x^3-x",grid=0); // plot

>frame; xgrid([-1,0,1]); ygrid(0): // tambahkan frame dan grid



Gambar 71: images/PLOT%202D_Diva%20Nagita_23030630024-073.png

Untuk judul plot dan label sumbu, lihat contoh berikut.

>plot2d("exp(x)",-1,1);

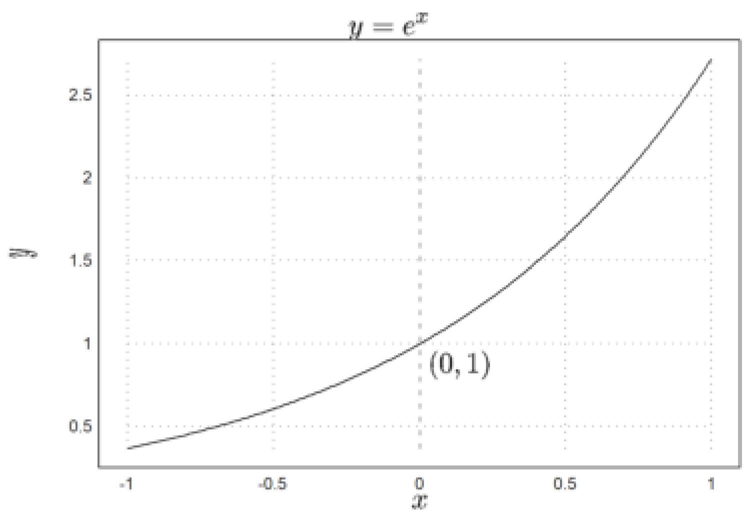
>textcolor(black); // setel warna teks menjadi hitam

>title(latex("y=e^x")); // judul di atas plot

>xlabel(latex("x")); // "x" untuk sumbu x

>ylabel(latex("y"),>vertical); // vertikal "y" untuk sumbu y

>label(latex("(0,1)"),0,1,color=blue): // memberi label sebuah titik

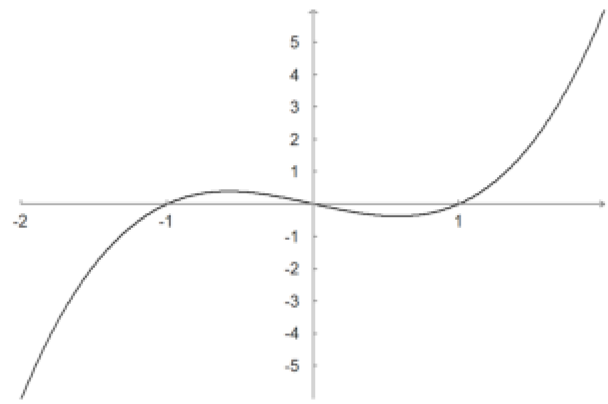


Gambar 72: images/PLOT%202D_Diva%20Nagita_23030630024-074.png

Sumbu dapat digambar secara terpisah dengan sumbu x() dan sumbu y().

>plot2d("x^3-x",<grid,<frame);

>xaxis(0,xx=-2:1,style="->"); yaxis(0,yy=-5:5,style="->"):



Gambar 73: images/PLOT%202D_Diva%20Nagita_23030630024-075.png

Teks pada plot dapat diatur dengan label(). Pada contoh berikut ini, “lc” berarti lower center. Ini mengatur posisi label relatif terhadap koordinat plot.

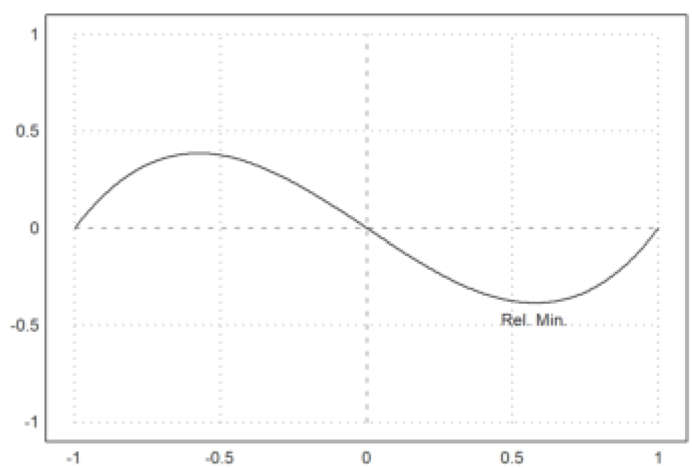
>function f(x) &= x^3-x

$$x^3 - x$$

>plot2d(f,-1,1,>square);

>x0=fmin(f,0,1); // compute point of minimum

>label(“Rel. Min.”,x0,f(x0),pos=“lc”): // tambahkan label di sana



Gambar 74: images/PLOT%202D_Diva%20Nagita_23030630024-076.png

Terdapat juga kotak teks.

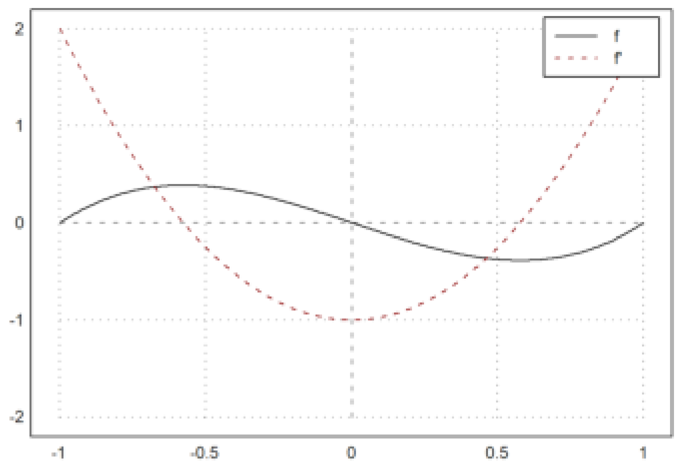
>plot2d(&f(x),-1,1,-2,2); // fungsi

>plot2d(&diff(f(x),x),>add,style=“–”,color=red); // turunan

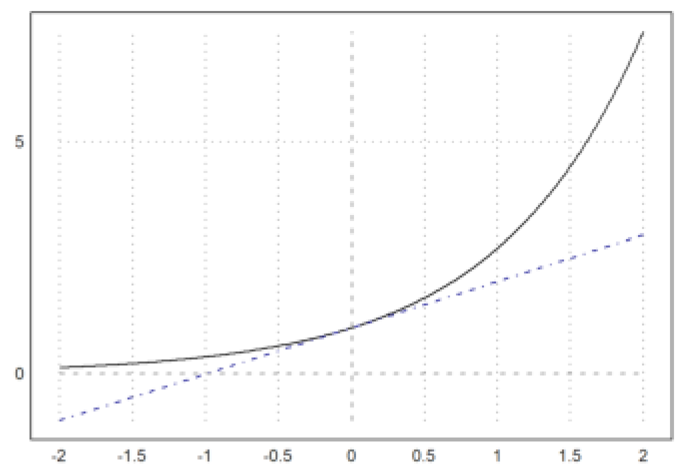
>labelbox([“f”,“f’”],[“-”,“–”],[black,red]): // kotak label

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

>gridstyle(“->”,color=gray,textcolor=gray,framecolor=gray); . . .
> plot2d(“x^3-x”,grid=1); . . .

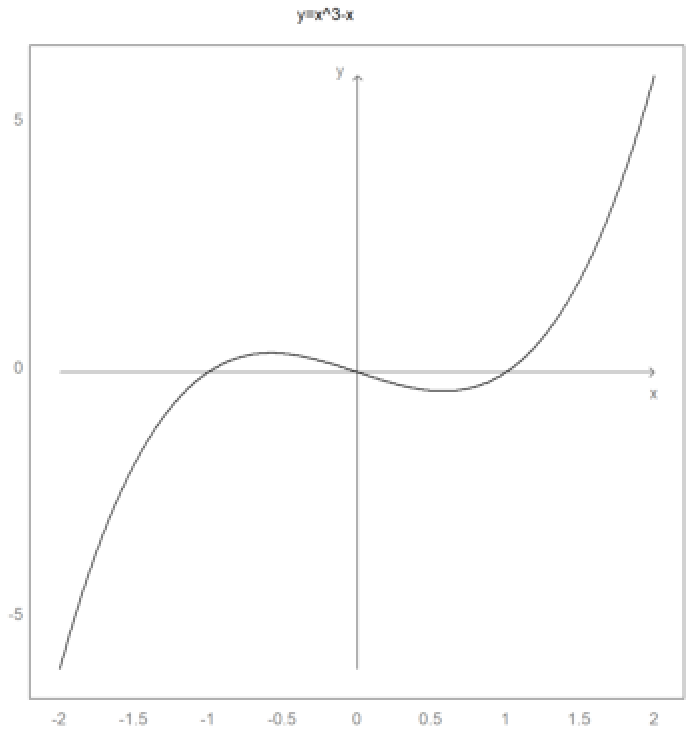


Gambar 75: images/PLOT%202D_Diva%20Nagita_23030630024-077.png



Gambar 76: images/PLOT%202D_Diva%20Nagita_23030630024-078.png

> settitle(“y=x^3-x”,color=black); . . .
> label(“x”,2,0,pos=“bc”,color=gray); . . .
> label(“y”,0,6,pos=“cl”,color=gray); . . .
> reset():



Gambar 77: images/PLOT%202D_Diva%20Nagita_23030630024-079.png

Untuk kontrol yang lebih besar lagi, sumbu x dan sumbu y dapat dilakukan secara manual.

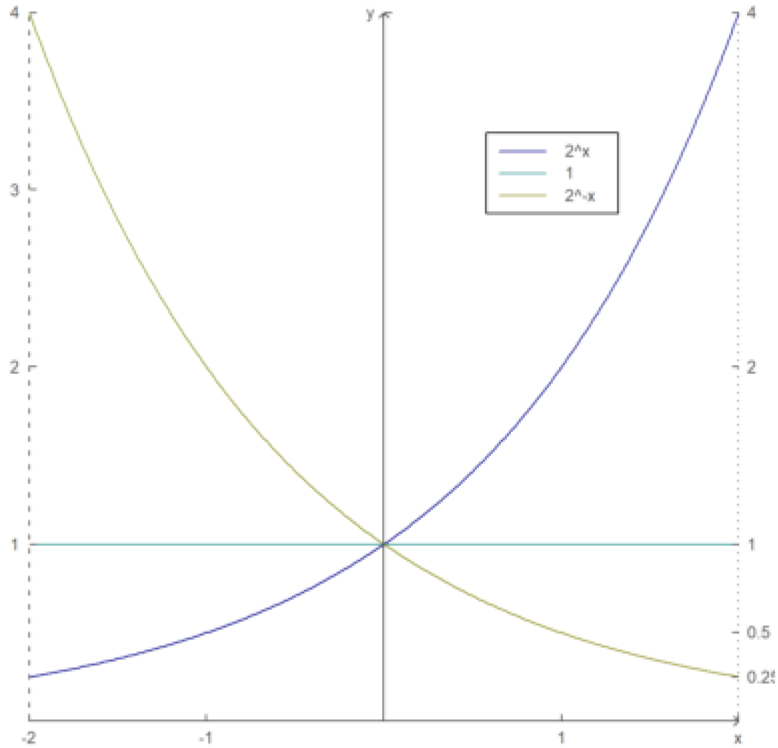
Perintah fullwindow() akan memperluas jendela plot karena kita tidak lagi membutuhkan tempat untuk label di luar jendela plot. Gunakan shrinkwindow() atau reset() untuk mengatur ulang ke default.

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

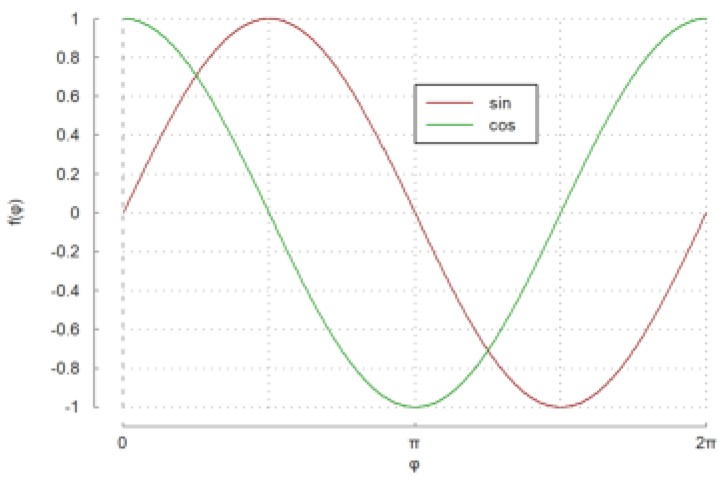
Berikut ini adalah contoh lain, di mana string Unicode digunakan dan sumbu di luar area plot.

>aspect(1.5);

>plot2d([“sin(x)”,“cos(x)”],0,2pi,color=[red,green],<grid,<frame); . . .
> xaxis(-1.1,(0:2)*pi,xt=[“0”,u”“,u”2”],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””); ylabel(u”f()“):



Gambar 78: images/PLOT%202D_Diva%20Nagita_23030630024-080.png



Gambar 79: images/PLOT%202D_Diva%20Nagita_23030630024-081.png

Memplot Data 2D

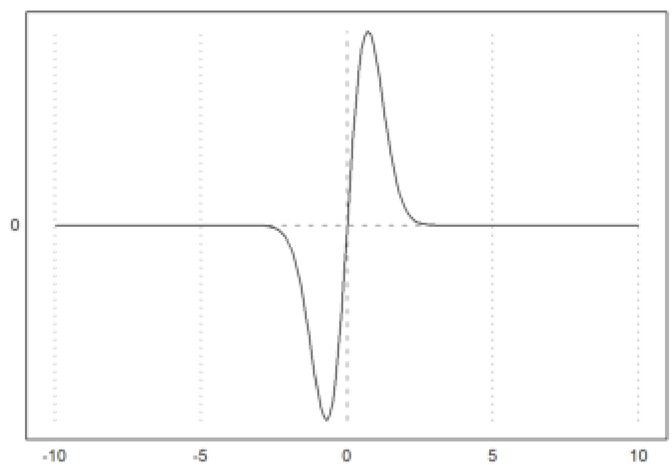
Jika x dan y adalah vektor data, data ini akan digunakan sebagai koordinat x dan y dari sebuah kurva. Dalam hal ini, a, b, c, dan d, atau radius r dapat ditentukan, atau jendela plot akan menyesuaikan secara otomatis dengan data. Sebagai alternatif, >square dapat diatur untuk mempertahankan rasio aspek persegi.

Memplot ekspresi hanyalah singkatan untuk 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 untuk diplot, secara default dengan evaluasi adaptif dari fungsi tersebut. Untuk plot titik, gunakan ">points", untuk garis dan titik campuran gunakan ">addpoints".

Namun Anda dapat memasukkan data secara langsung. + Gunakan vektor baris untuk x dan y untuk satu fungsi. + Matriks untuk x dan y diplot baris demi baris.

Berikut adalah contoh dengan satu baris untuk x dan y.

>x=-10:0.1:10; y=exp(-x^2)*x; plot2d(x,y):



Gambar 80: images/PLOT%202D_Diva%20Nagita_23030630024-082.png

Data juga dapat diplot sebagai titik. Gunakan poin=true untuk ini. Plot ini bekerja seperti poligon, namun hanya menggambar sudut-sudutnya saja.

- style = "...": Pilih dari "[]", "<>", "o", ".", "..", "+", "*", "[]#", "<>#", "o#", "..#", "#", "|".

Untuk memplot kumpulan titik, gunakan >titik. Jika warna adalah sebuah vektor warna, setiap titik mendapatkan warna yang berbeda. Untuk sebuah matriks koordinat dan vektor kolom, warna berlaku pada baris-baris matriks.

Parameter >addpoints menambahkan titik-titik pada 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="."); // garis

>plot2d(xdata,ydata,>points,>add,style="o"): // menambahkan titik

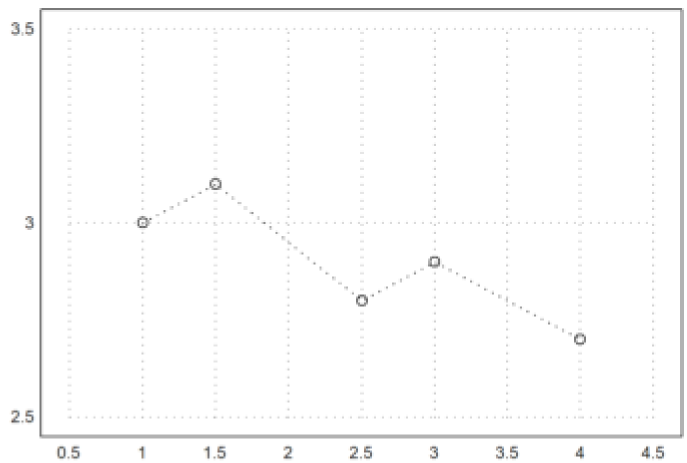
>p=polyfit(xdata,ydata,1); // mendapatkan garis regresi

>plot2d("polyval(p,x)",>add,color=red): //menambahkan plot garis

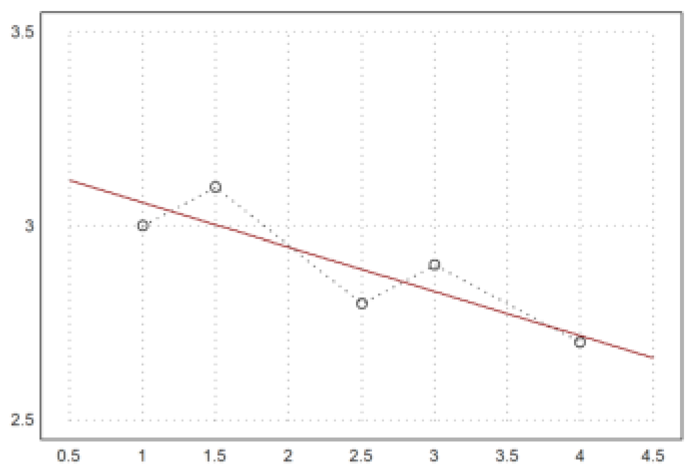
Menggambar Daerah Yang Dibatasi Kurva

Plot data sebenarnya adalah poligon. Kita juga dapat memplot kurva atau kurva yang terisi.

- filled=true mengisi plot.
- style = "...": Pilih dari "#", "/", "", "/".



Gambar 81: images/PLOT%202D_Diva%20Nagita_23030630024-083.png



Gambar 82: images/PLOT%202D_Diva%20Nagita_23030630024-084.png

- fillcolor: Lihat di atas untuk warna yang tersedia.

Warna isian ditentukan oleh argumen “fillcolor”, dan pada pilihan <outline mencegah menggambar batas untuk semua gaya kecuali gaya default.

>t=linspace(0,2pi,1000); // parameter untuk kurva

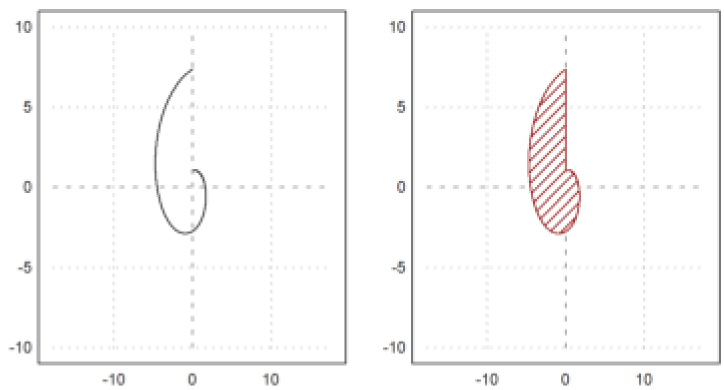
>x=sin(t)*exp(t/pi); y=cos(t)*exp(t/pi); // x(t) dan y(t)

>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

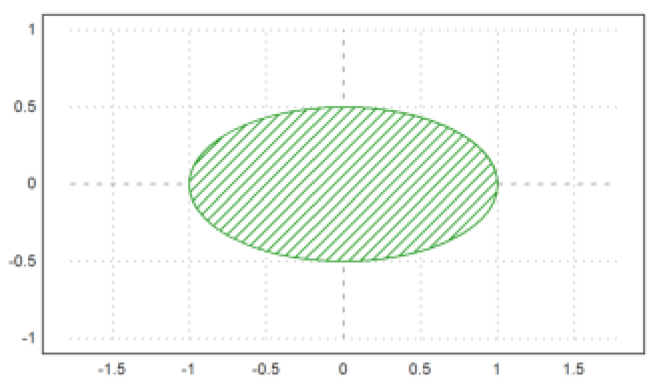
>figure(0):



Gambar 83: images/PLOT%202D_Diva%20Nagita_23030630024-085.png

Pada contoh berikut ini, kami memplot elips terisi dan dua segi enam terisi menggunakan kurva tertutup dengan 6 titik dengan gaya isian yang berbeda.

>x=linspace(0,2pi,1000); plot2d(sin(x),cos(x)*0.5,r=1,>filled,style="/"):



Gambar 84: images/PLOT%202D_Diva%20Nagita_23030630024-086.png

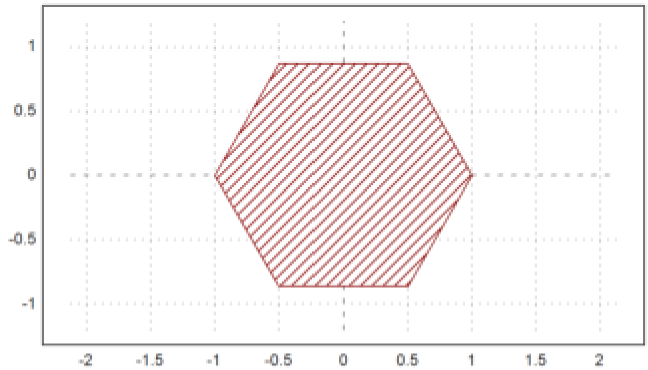
>t=linspace(0,2pi,6); . . .
> plot2d(cos(t),sin(t),>filled,style="/",fillcolor=red,r=1.2):

>t=linspace(0,2pi,6); plot2d(cos(t),sin(t),>filled,style="#"):

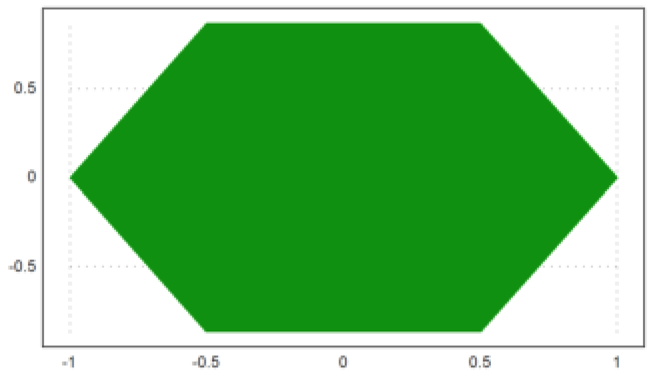
Contoh lainnya adalah septagon, yang kita buat dengan 7 titik pada lingkaran satuan.

>t=linspace(0,2pi,7); . . .
> plot2d(cos(t),sin(t),r=1,>filled,style="/",fillcolor=red):

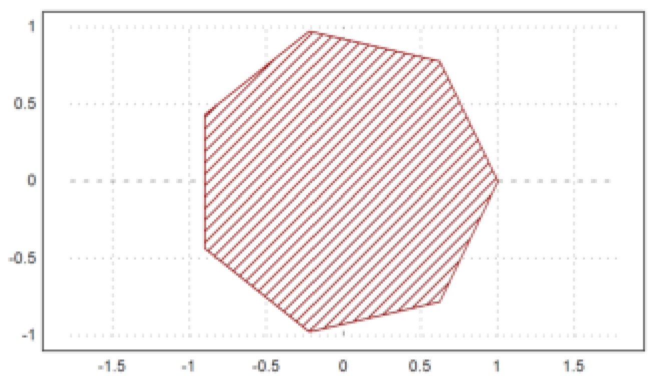
Berikut ini adalah himpunan nilai maksimal dari empat kondisi linier yang kurang dari atau sama dengan 3. Ini adalah A[k].v<=3 untuk semua barisan A. Untuk mendapatkan sudut-sudut yang bagus, kita menggunakan n yang relatif besar.



Gambar 85: images/PLOT%202D_Diva%20Nagita_23030630024-087.png



Gambar 86: images/PLOT%202D_Diva%20Nagita_23030630024-088.png

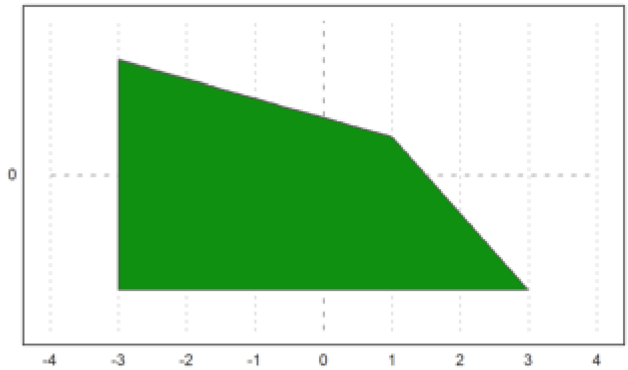


Gambar 87: images/PLOT%202D_Diva%20Nagita_23030630024-089.png

>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):



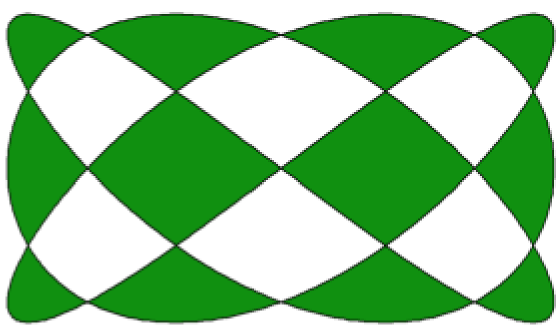
Gambar 88: images/PLOT%202D_Diva%20Nagita_23030630024-090.png

Poin utama dari bahasa matriks adalah bahwa bahasa ini memungkinkan untuk menghasilkan tabel fungsi dengan mudah.

>t=linspace(0,2pi,1000); x=cos(3*t); y=sin(4*t);

Kita sekarang memiliki vektor nilai x dan y. plot2d() dapat memplot nilai-nilai ini sebagai sebuah kurva yang menghubungkan titik-titik. Plot dapat diisi. Dalam kasus ini, hal ini memberikan hasil yang bagus karena aturan lilitan, yang digunakan untuk mengisi.

>plot2d(x,y,<grid,<frame,>filled):



Gambar 89: images/PLOT%202D_Diva%20Nagita_23030630024-091.png

Vektor interval diplot terhadap nilai x sebagai wilayah yang terisi

antara nilai bawah dan atas interval.

Hal ini dapat berguna untuk memplot kesalahan perhitungan. Tapi itu bisa

juga dapat digunakan untuk memplot kesalahan statistik.

>t=0:0.1:1; ...

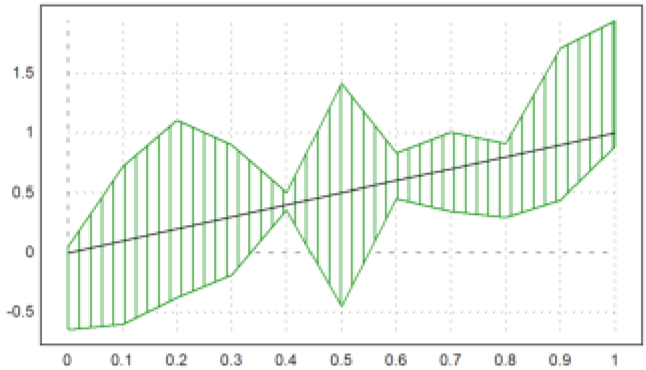
> plot2d(t,interval(t-random(size(t)),t+random(size(t))),style="|"); ...

> plot2d(t,t,add=true):

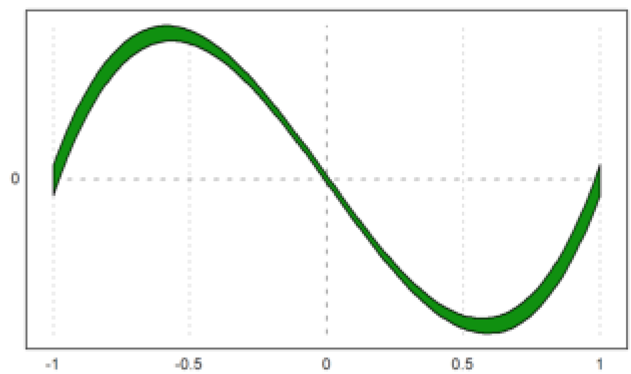
Jika x adalah vektor yang diurutkan, dan y adalah vektor interval, maka plot2d akan memplot rentang interval yang terisi pada bidang, gaya isian sama dengan gaya poligon.

>t=-1:0.01:1; x=$_{t-0.01,t+0.01}$; y=x^3-x;

>plot2d(t,y):



Gambar 90: images/PLOT%202D_Diva%20Nagita_23030630024-092.png

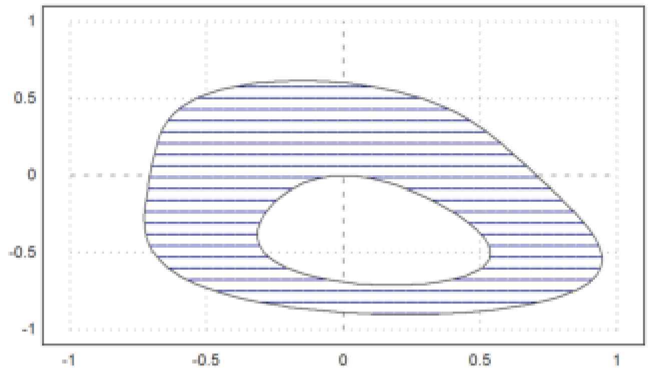


Gambar 91: images/PLOT%202D_Diva%20Nagita_23030630024-093.png

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”; // mendefinisikan ekspresi f(x,y)

>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <= 1

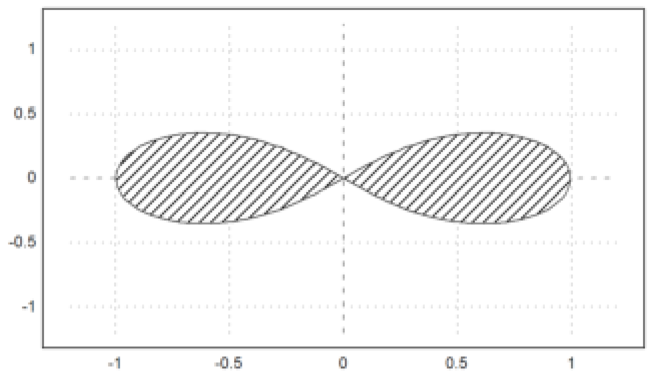


Gambar 92: images/PLOT%202D_Diva%20Nagita_23030630024-094.png

Kita juga dapat mengisi rentang nilai seperti

$$-1 \leq (x^2 + y^2)^2 - x^2 + y^2 \leq 0.$$

>plot2d("(x^{2+y}2)$^{2-x}$2+y^2",r=1.2,level=[-1;0],style="/"):



Gambar 93: images/PLOT%202D_Diva%20Nagita_23030630024-096.png

>plot2d("cos(x)","sin(x)^3",xmin=0,xmax=2pi,>filled,style="/"):

Grafik Fungsi Parametrik

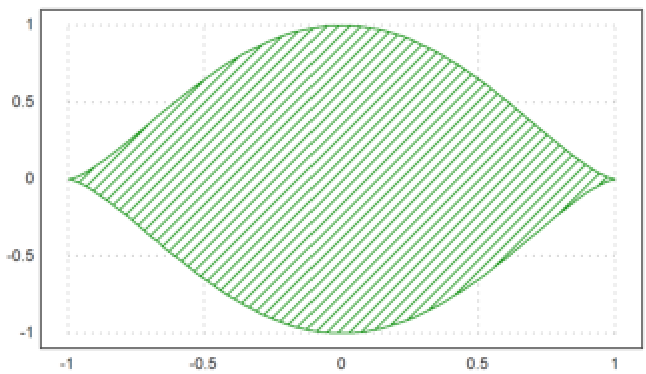
Nilai x tidak perlu diurutkan. (x,y) hanya menggambarkan sebuah kurva. Jika x diurutkan, kurva tersebut adalah grafik fungsi.

Pada contoh berikut, kita memplot spiral

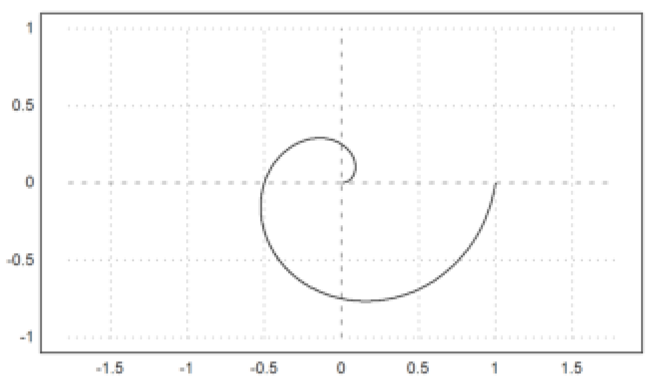
$$\gamma(t) = t \cdot (\cos(2\pi t), \sin(2\pi t))$$

Kita mungkin perlu menggunakan sangat banyak titik untuk tampilan yang halus atau fungsi adaptive() untuk mengevaluasi ekspresi (lihat fungsi adaptive() untuk lebih jelasnya).

>t=linspace(0,1,1000); ...
> plot2d(t*cos(2*pi*t),t*sin(2*pi*t),r=1):



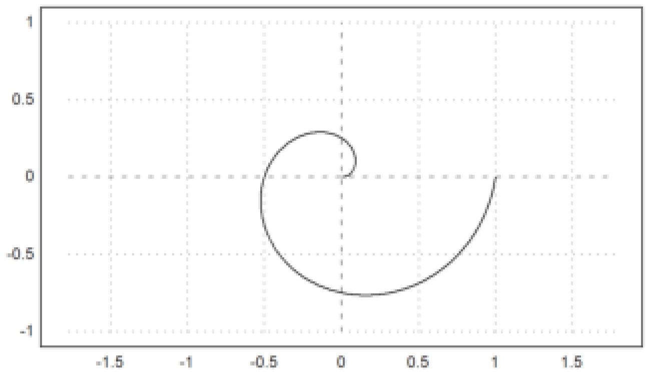
Gambar 94: images/PLOT%202D_Diva%20Nagita_23030630024-097.png



Gambar 95: images/PLOT%202D_Diva%20Nagita_23030630024-099.png

Sebagai alternatif, Anda dapat menggunakan dua ekspresi untuk kurva. Berikut ini memplot kurva yang sama seperti di atas.

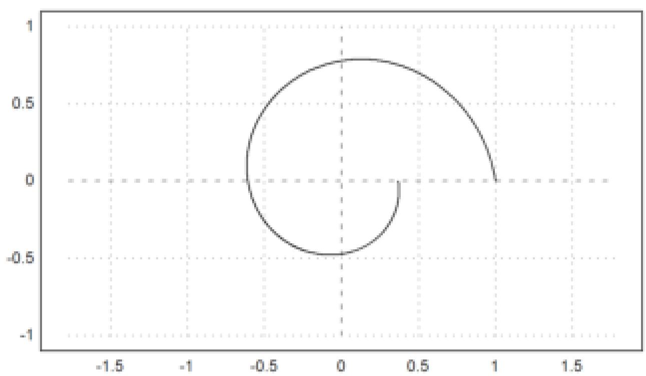
>plot2d("x*cos(2*pi*x)","x*sin(2*pi*x)",xmin=0,xmax=1,r=1):



Gambar 96: images/PLOT%202D_Diva%20Nagita_23030630024-100.png

>t=linspace(0,1,1000); r=exp(-t); x=r*cos(2pi*t); y=r*sin(2pi*t);

>plot2d(x,y,r=1):



Gambar 97: images/PLOT%202D_Diva%20Nagita_23030630024-101.png

Pada contoh berikut ini, kami memplot kurva

$$\gamma(t) = (r(t)\cos(t), r(t)\sin(t))$$

dengan

$$r(t) = 1 + \frac{\sin(3t)}{2}.$$

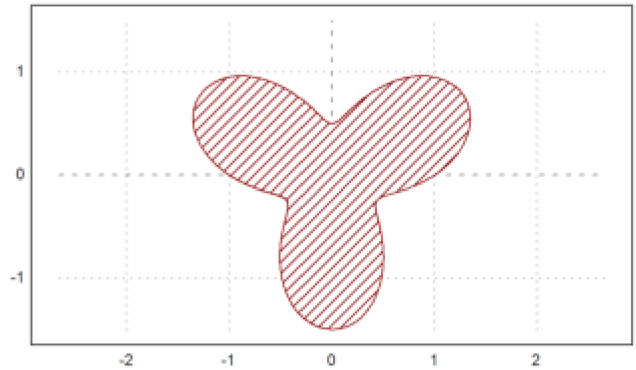
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t); ...

> plot2d(x,y,>filled,fillcolor=red,style="/",r=1.5):

Menggambar Grafik Bilangan Kompleks

Sebuah deretan bilangan kompleks juga dapat diplot. Kemudian titik-titik kisi akan dihubungkan. Jika sejumlah garis kisi ditentukan (atau vektor 1x2 garis kisi) pada argumen cgrid, hanya garis-garis kisi tersebut yang akan terlihat.

Matriks bilangan kompleks akan secara otomatis diplot sebagai sebuah grid pada bidang kompleks.

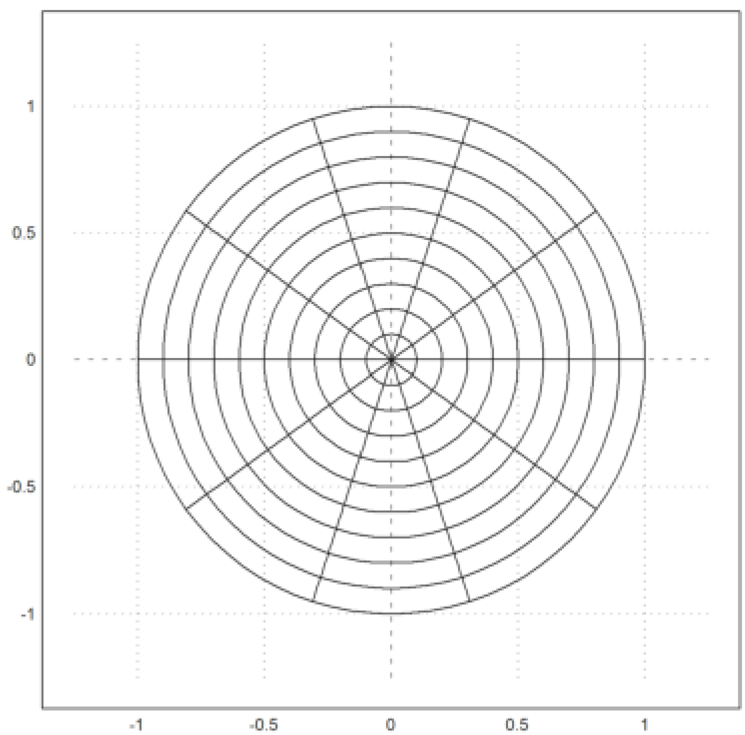


Gambar 98: images/PLOT%202D_Diva%20Nagita_23030630024-104.png

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):



Gambar 99: images/PLOT%202D_Diva%20Nagita_23030630024-105.png

>aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)'; z=r*exp(I*a);

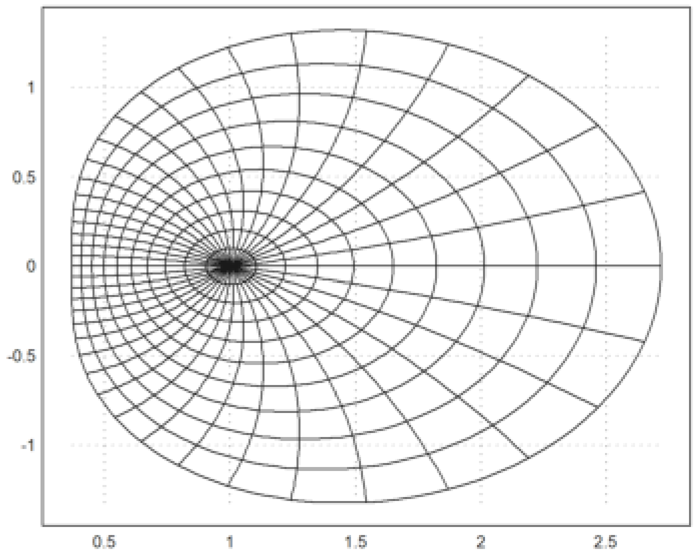
>plot2d(exp(z),cgrid=[40,10]):

>r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=r*exp(I*a);

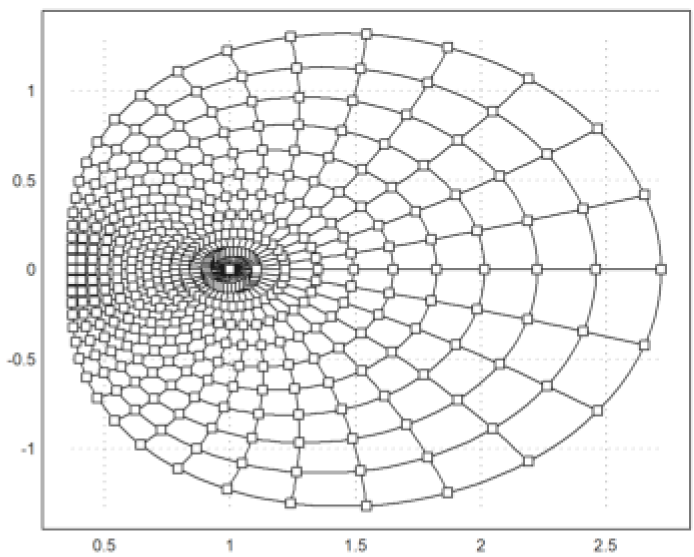
>plot2d(exp(z),>points,>add):

Vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian nyata dan bagian imajiner.

Pada contoh, kami memplot lingkaran satuan dengan



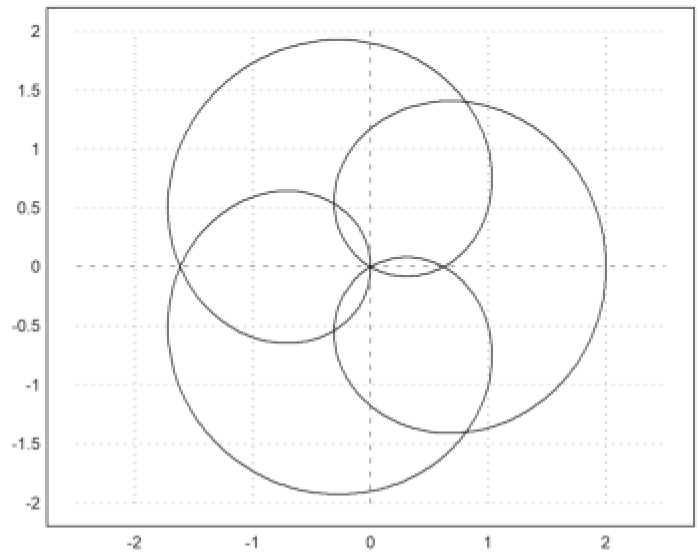
Gambar 100: images/PLOT%202D_Diva%20Nagita_23030630024-106.png



Gambar 101: images/PLOT%202D_Diva%20Nagita_23030630024-107.png

$$\gamma(t) = e^{it}$$

>t=linspace(0,2pi,1000); . . .
> plot2d(exp(I*t)+exp(4*I*t),r=2):



Gambar 102: images/PLOT%202D_Diva%20Nagita_23030630024-109.png

Plot Statistik

Terdapat banyak fungsi yang dikhususkan untuk plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

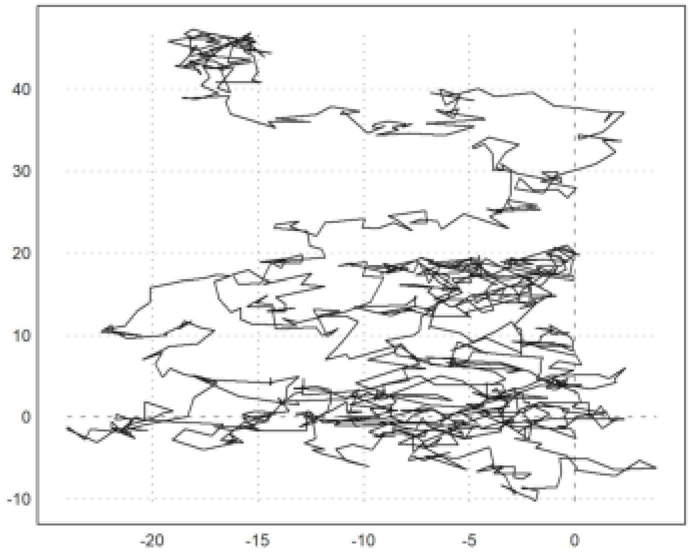
Jumlah kumulatif dari nilai berdistribusi normal 0-1 menghasilkan jalan acak.

>plot2d(cumsum(randnormal(1,1000))):

Gambar 103: images/PLOT%202D_Diva%20Nagita_23030630024-110.png

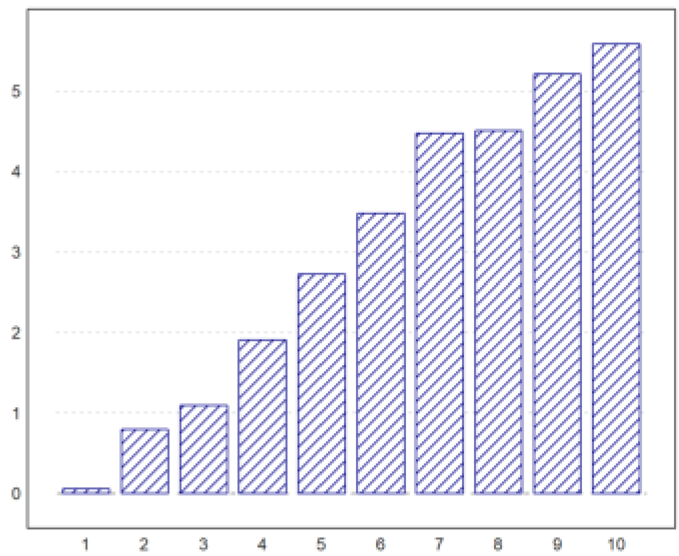
Dengan menggunakan dua baris, ini menunjukkan jalan kaki dalam dua dimensi.

>X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]):



Gambar 104: images/PLOT%202D_Diva%20Nagita_23030630024-111.png

>columnsplot(cumsum(random(10)),style="/",color=blue):



Gambar 105: images/PLOT%202D_Diva%20Nagita_23030630024-112.png

Ini juga dapat menampilkan string sebagai label.

>months=["Jan","Feb","Mar","Apr","May","Jun", . . .
> "Jul","Aug","Sep","Oct","Nov","Dec"];

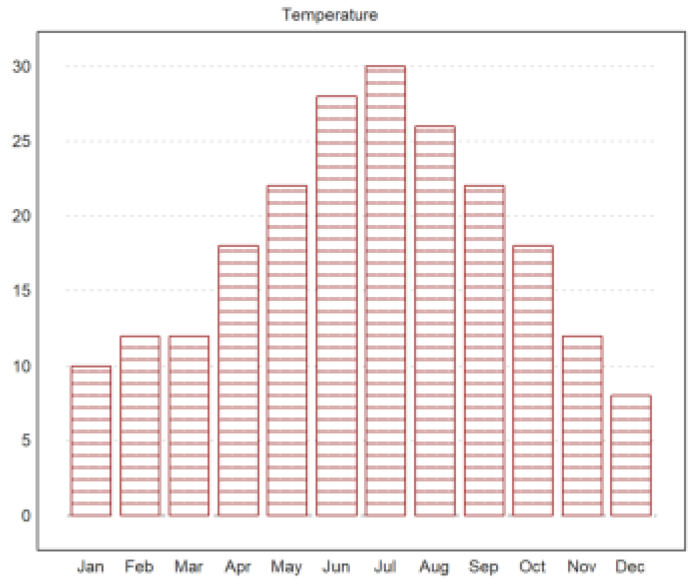
>values=[10,12,12,18,22,28,30,26,22,18,12,8];

>columnsplot(values,lab=months,color=red,style="-");

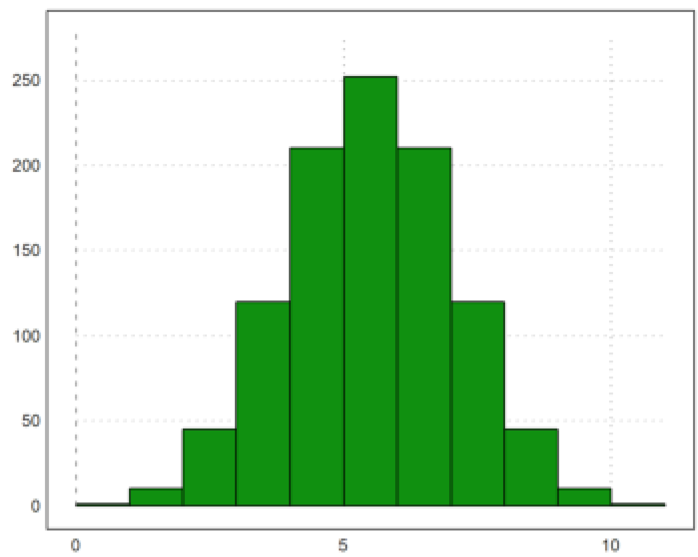
>title("Temperature"):

>k=0:10;

>plot2d(k,bin(10,k),>bar):

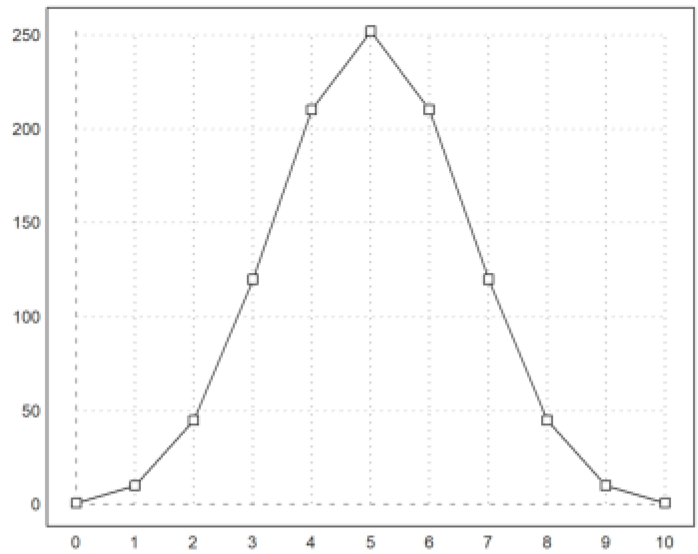


Gambar 106: images/PLOT%202D_Diva%20Nagita_23030630024-113.png



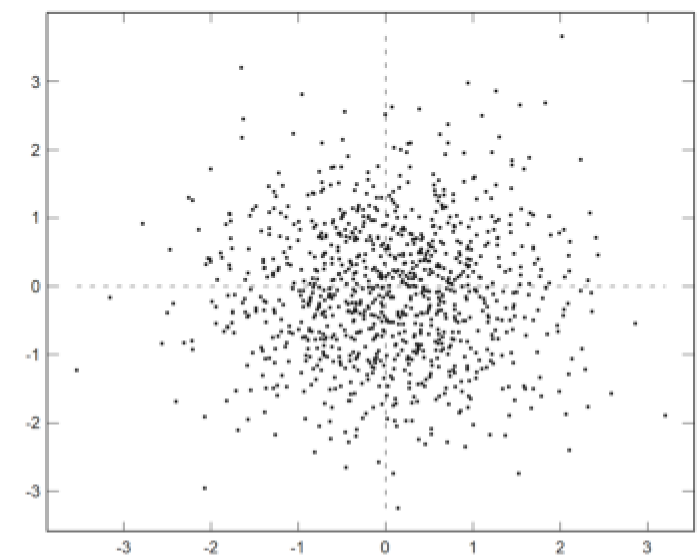
Gambar 107: images/PLOT%202D_Diva%20Nagita_23030630024-114.png

>plot2d(k,bin(10,k)); plot2d(k,bin(10,k),>points,>add):



Gambar 108: images/PLOT%202D_Diva%20Nagita_23030630024-115.png

>plot2d(normal(1000),normal(1000),>points,grid=6,style="..");



Gambar 109: images/PLOT%202D_Diva%20Nagita_23030630024-116.png

>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.

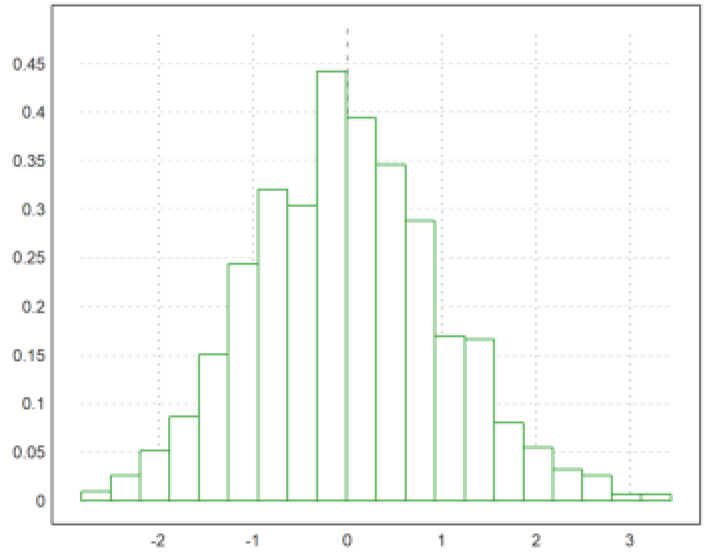
>w=randexponential(1,1000); // exponential distribution

>plot2d(w,>distribution): // or distribution=n with n intervals

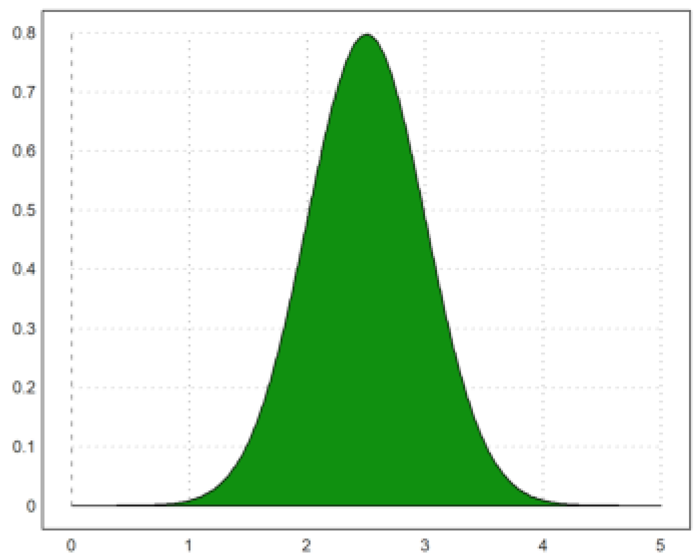
Atau Anda dapat menghitung distribusi dari data dan memplot hasilnya dengan >bar di plot3d, atau dengan plot kolom.

>w=normal(1000); // 0-1-normal distribution

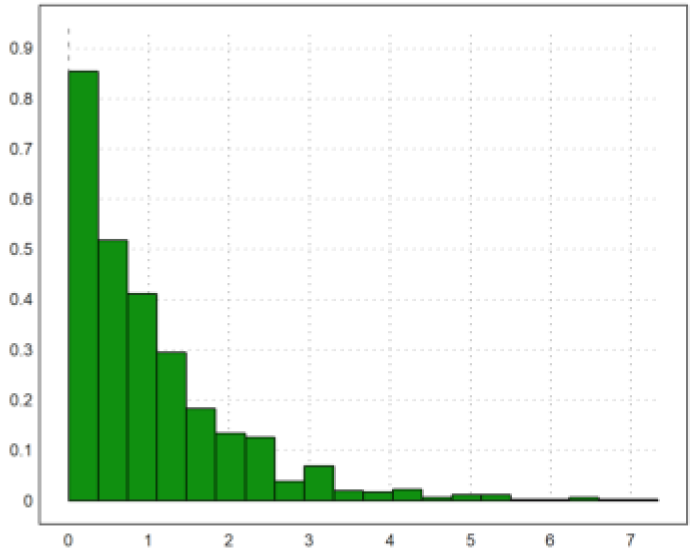
>{x,y}=histo(w,10,v=[-6,-4,-2,-1,0,1,2,4,6]); // interval bounds v



Gambar 110: images/PLOT%202D_Diva%20Nagita_23030630024-117.png

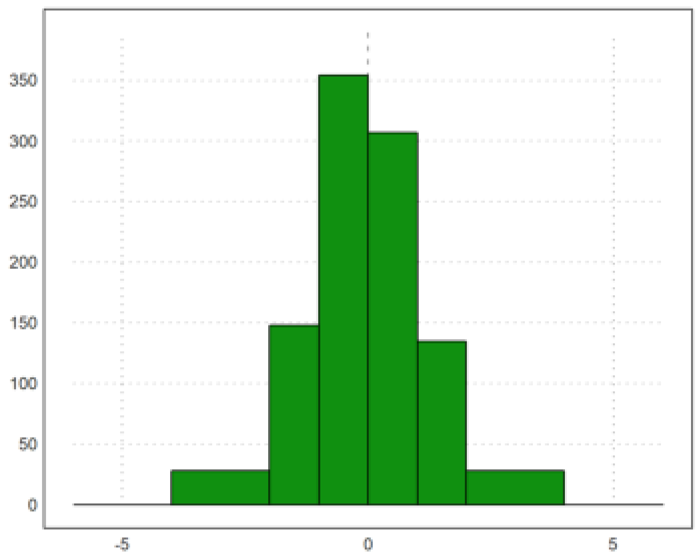


Gambar 111: images/PLOT%202D_Diva%20Nagita_23030630024-118.png



Gambar 112: images/PLOT%202D_Diva%20Nagita_23030630024-119.png

>plot2d(x,y,>bar):



Gambar 113: images/PLOT%202D_Diva%20Nagita_23030630024-120.png

Fungsi statplot() menetapkan gaya dengan string sederhana.

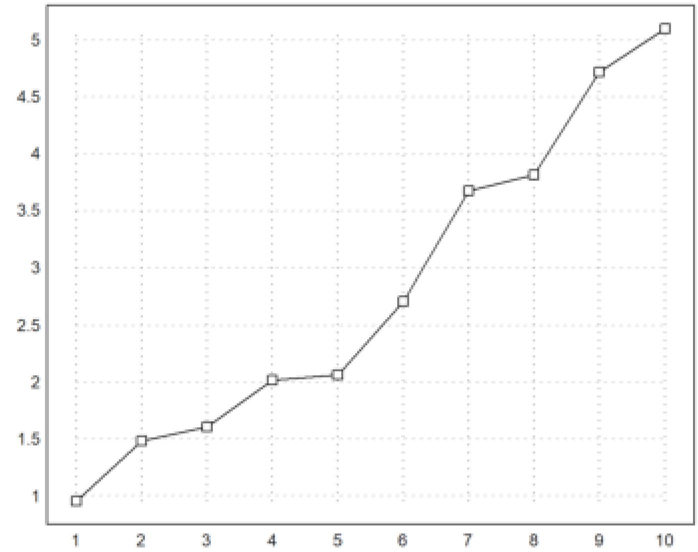
>statplot(1:10,cumsum(random(10)),"b"):

>n=10; i=0:n; . . .
> plot2d(i,bin(n,i)/2^n,a=0,b=10,c=0,d=0.3); . . .
> plot2d(i,bin(n,i)/2^n,points=true,style="ow",add=true,color=blue):

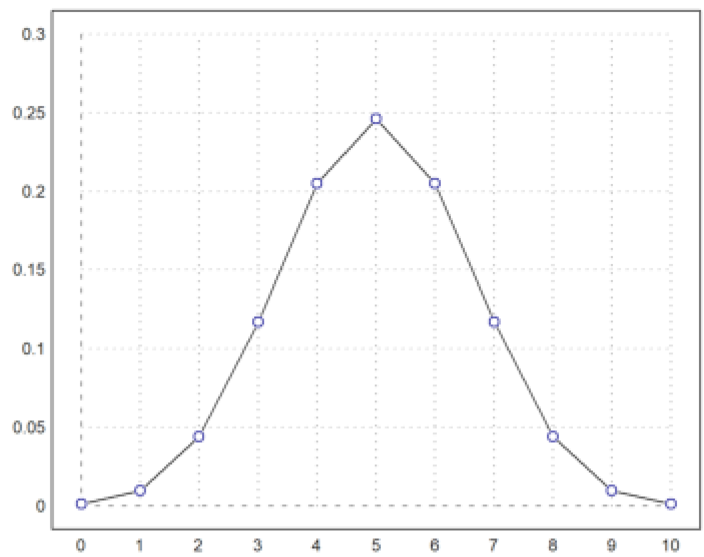
Selain itu, data dapat diplot sebagai batang. Dalam hal ini, x harus diurutkan dan satu elemen lebih panjang dari y. Batang akan memanjang dari x[i] ke x[i+1] dengan nilai y[i]. Jika x memiliki ukuran yang sama dengan y, maka x akan diperpanjang satu elemen dengan jarak terakhir.

Gaya isian dapat digunakan seperti di atas.

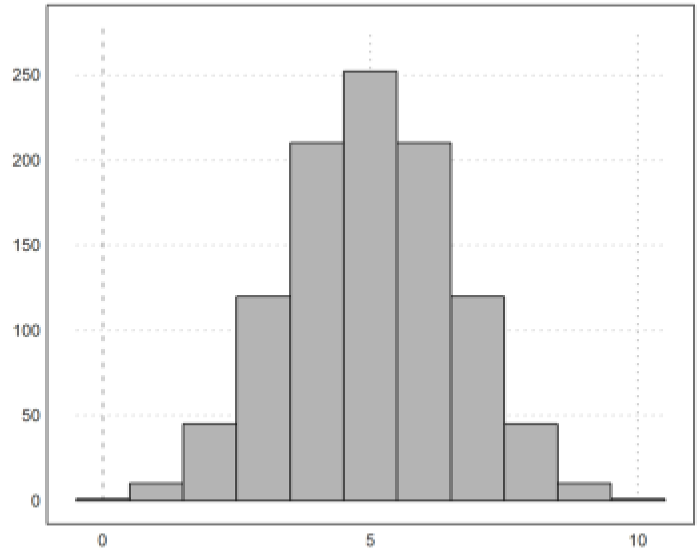
>n=10; k=bin(n,0:n); . . .
> plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray):



Gambar 114: images/PLOT%202D_Diva%20Nagita_23030630024-121.png



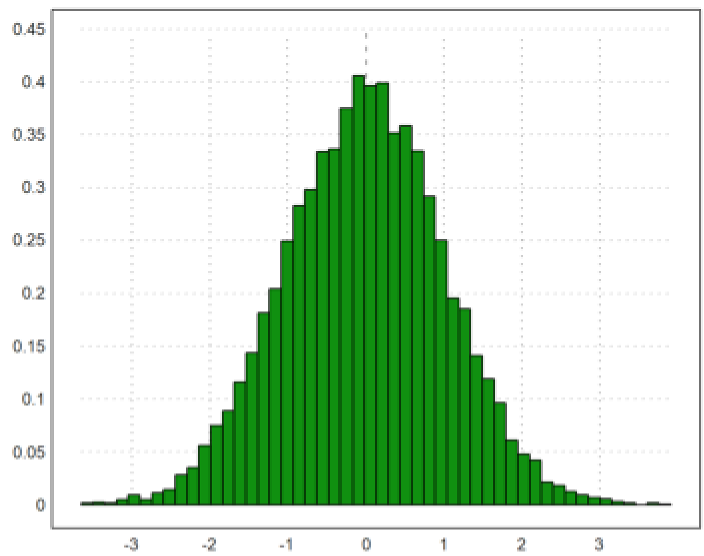
Gambar 115: images/PLOT%202D_Diva%20Nagita_23030630024-122.png



Gambar 116: images/PLOT%202D_Diva%20Nagita_23030630024-123.png

Data untuk plot batang (batang = 1) dan histogram (histogram = 1) dapat diberikan secara eksplisit dalam xv dan yv, atau dapat dihitung dari distribusi empiris dalam xv dengan >distribusi (atau distribusi = n). Histogram dari nilai xv akan dihitung secara otomatis dengan >histogram. Jika >even ditentukan, nilai xv akan dihitung dalam interval bilangan bulat.

>plot2d(normal(10000),distribution=50):



Gambar 117: images/PLOT%202D_Diva%20Nagita_23030630024-124.png

>k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m,>bar):

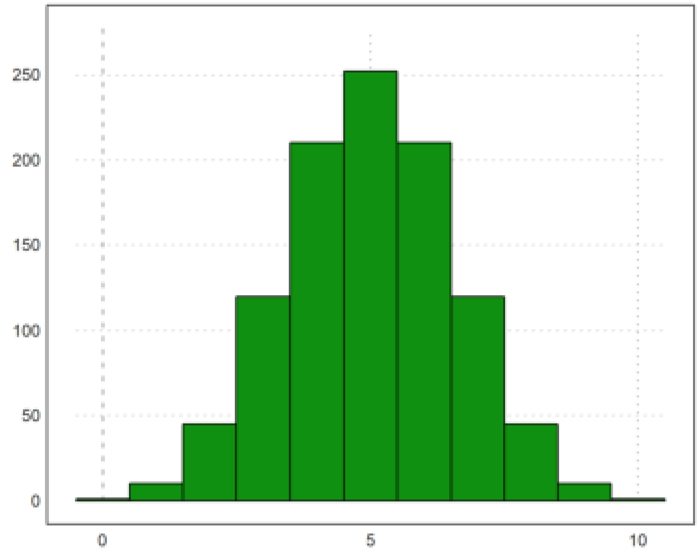
>columnsplot(m,k):

>plot2d(random(600)*6,histogram=6):

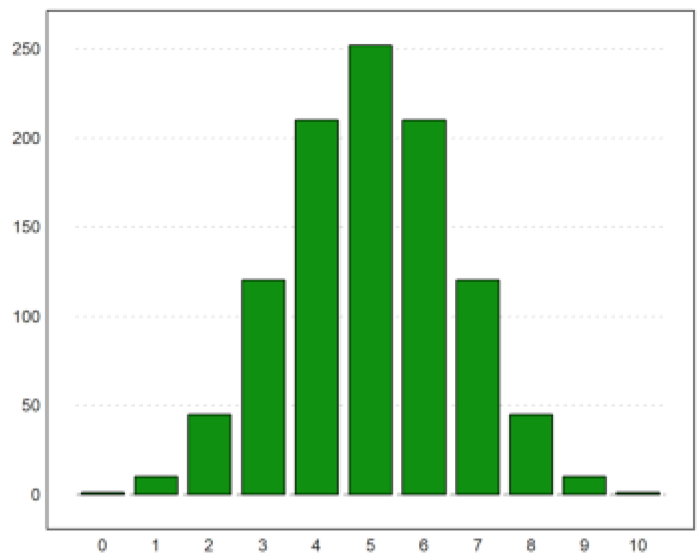
Untuk distribusi, ada parameter distribution=n, yang menghitung nilai secara otomatis dan mencetak distribusi relatif dengan n sub-interval.

>plot2d(normal(1,1000),distribution=10,style="\/"):

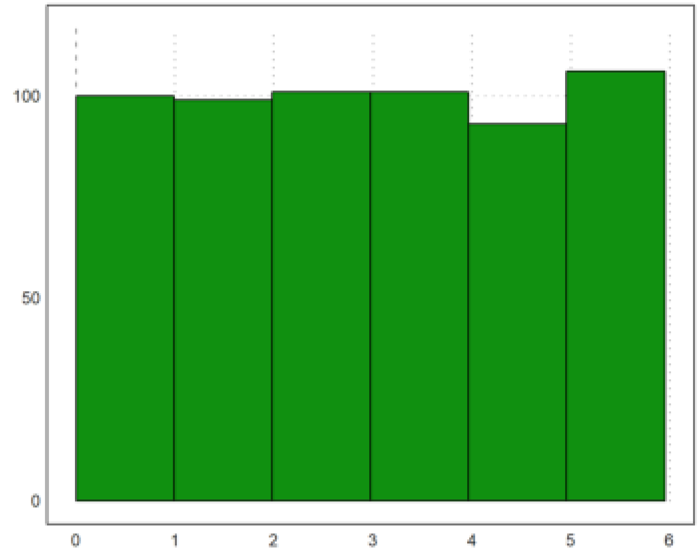
Dengan parameter even=true, ini akan menggunakan interval bilangan bulat.



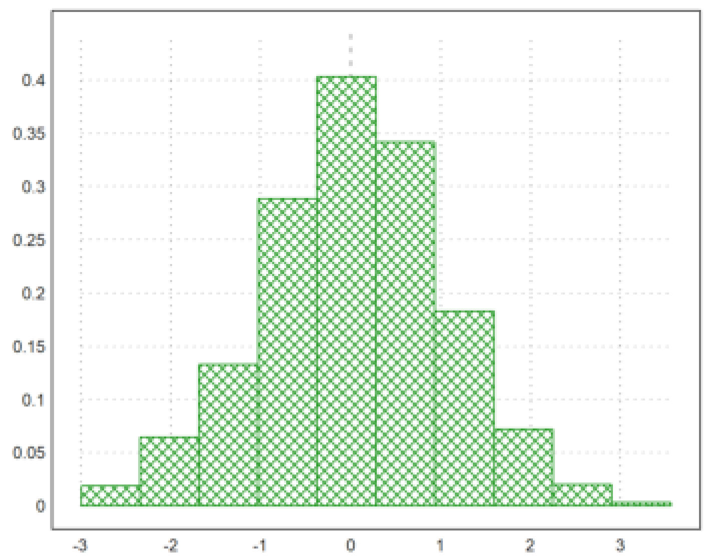
Gambar 118: images/PLOT%202D_Diva%20Nagita_23030630024-125.png



Gambar 119: images/PLOT%202D_Diva%20Nagita_23030630024-126.png

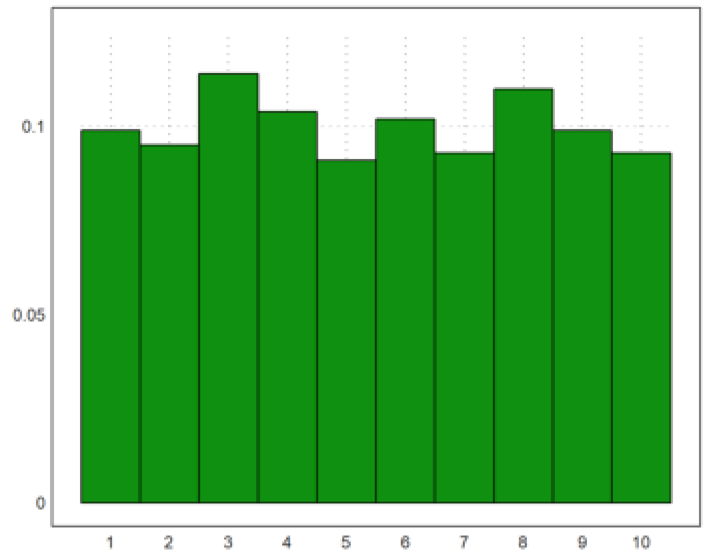


Gambar 120: images/PLOT%202D_Diva%20Nagita_23030630024-127.png



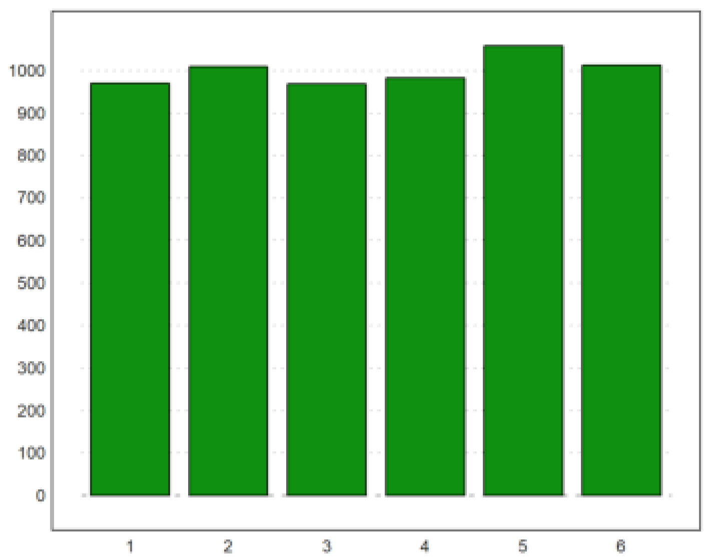
Gambar 121: images/PLOT%202D_Diva%20Nagita_23030630024-128.png

>plot2d(intrandom(1,1000,10),distribution=10,even=true):



Gambar 122: images/PLOT%202D_Diva%20Nagita_23030630024-129.png

Perhatikan bahwa ada banyak plot statistik yang mungkin berguna. Lihatlah tutorial tentang statistik.
>columnsplot(getmultiplicities(1:6,intrandom(1,6000,6))):

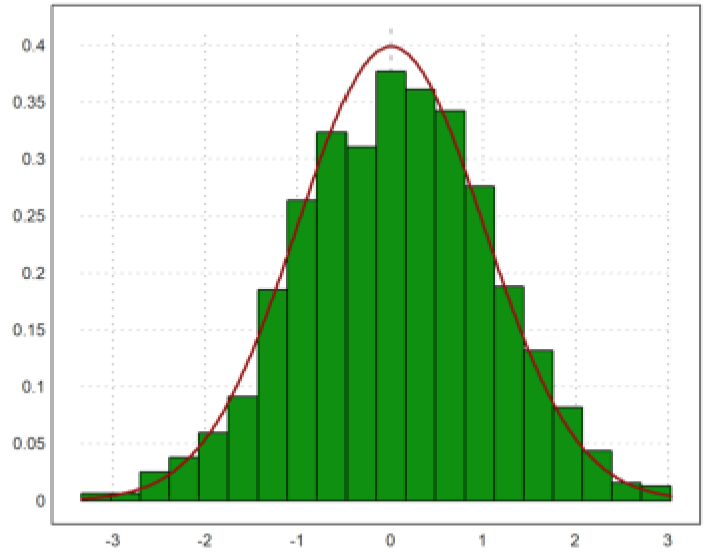


Gambar 123: images/PLOT%202D_Diva%20Nagita_23030630024-130.png

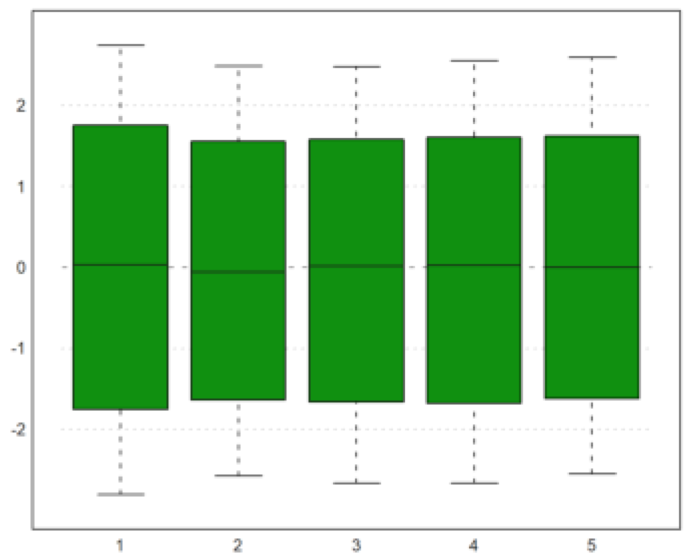
>plot2d(normal(1,1000),>distribution); . . .
> plot2d("qnormal(x)",color=red,thickness=2,>add):

Ada juga banyak plot khusus untuk statistik. Boxplot menunjukkan kuartil dari distribusi ini dan banyak pencilan. Menurut definisi, pencilan dalam boxplot adalah data yang melebihi 1,5 kali kisaran 50% tengah plot.

>M=normal(5,1000); boxplot(quartiles(M)):



Gambar 124: images/PLOT%202D_Diva%20Nagita_23030630024-131.png



Gambar 125: images/PLOT%202D_Diva%20Nagita_23030630024-132.png

Fungsi Implisit

Plot implisit menunjukkan garis level yang menyelesaikan f(x,y)=level, di mana “level” dapat berupa nilai tunggal atau vektor nilai. Jika level = “auto”, akan ada nc garis level, yang akan menyebar di antara minimum dan maksimum fungsi secara merata. Warna yang lebih gelap atau lebih terang dapat ditambahkan dengan >hue untuk mengindikasikan nilai fungsi. Untuk fungsi implisit, xv haruslah sebuah fungsi atau ekspresi dari parameter x dan y, atau, sebagai alternatif, xv dapat berupa matriks nilai.

Euler dapat menandai garis level

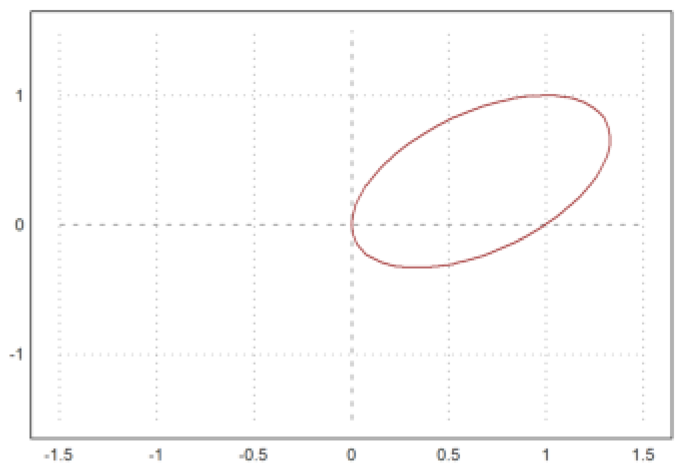
$$f(x, y) = c$$

dari fungsi apa pun.

Untuk menggambar himpunan f(x,y) = c untuk satu atau lebih konstanta c, Anda bisa menggunakan plot2d() dengan plot implisitnya pada bidang. Parameter untuk c adalah level = c, di mana c dapat berupa vektor garis level. Sebagai tambahan, sebuah skema warna dapat digambar pada latar belakang untuk mengindikasikan nilai fungsi untuk setiap titik pada plot. Parameter “n” menentukan kehalusan plot.

>aspect(1.5);

>plot2d(“x^{2+y}2-x*y-x”,r=1.5,level=0,contourcolor=red):



Gambar 126: images/PLOT%202D_Diva%20Nagita_23030630024-134.png

>expr := “2*x$^{2+x*y+3*y}$4+y”; // define an expression f(x,y)

>plot2d(expr,level=0): // Solutions of f(x,y)=0

>plot2d(expr,level=0:0.5:20,>hue,contourcolor=white,n=200): // nice

>plot2d(expr,level=0:0.5:20,>hue,>spectral,n=200,grid=4): //nicer

Hal ini juga berlaku untuk plot data. Tetapi Anda harus menentukan rentang untuk label sumbu.

>x=-2:0.05:1; y=x’; z=expr(x,y);

>plot2d(z,level=0,a=-1,b=2,c=-2,d=1,>hue):

>plot2d(“x^{3-y}2”,>contour,>hue,>spectral):

>plot2d(“x^{3-y}2”,level=0,contourwidth=3,>add,contourcolor=red):

>z=z+normal(size(z))*0.2;

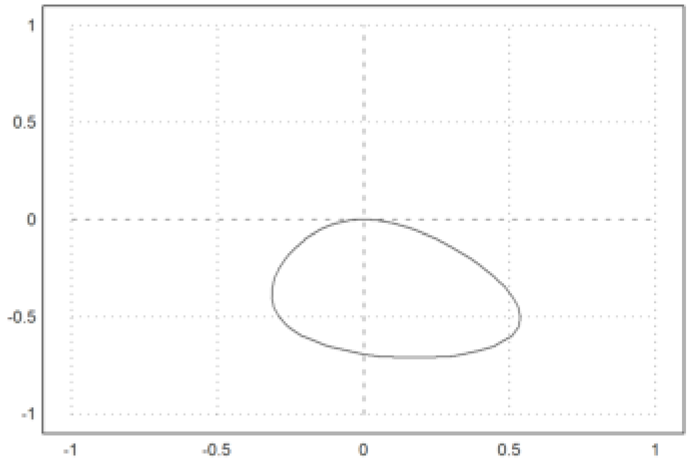
>plot2d(z,level=0.5,a=-1,b=2,c=-2,d=1):

>plot2d(expr,level=[0:0.2:5;0.05:0.2:5.05],color=lightgray):

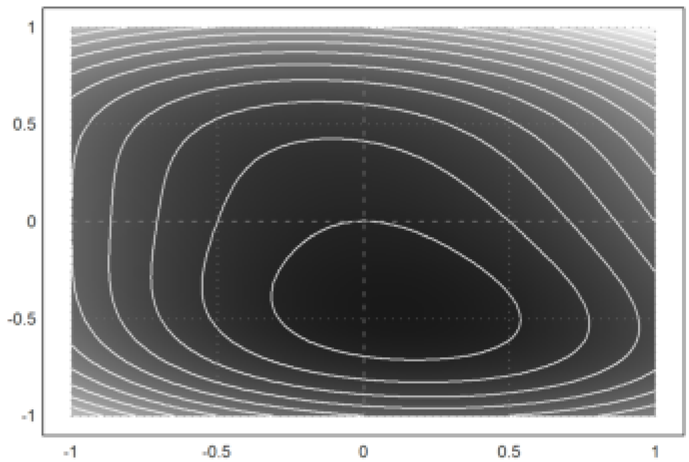
>plot2d(“x^{2+y}3+x*y”,level=1,r=4,n=100):

>plot2d(“x^{2+2*y}2-x*y”,level=0:0.1:10,n=100,contourcolor=white,>hue):

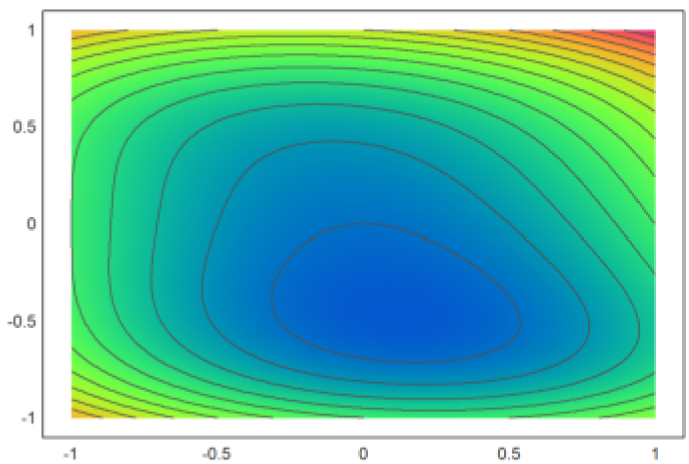
Dimungkinkan juga untuk mengisi set



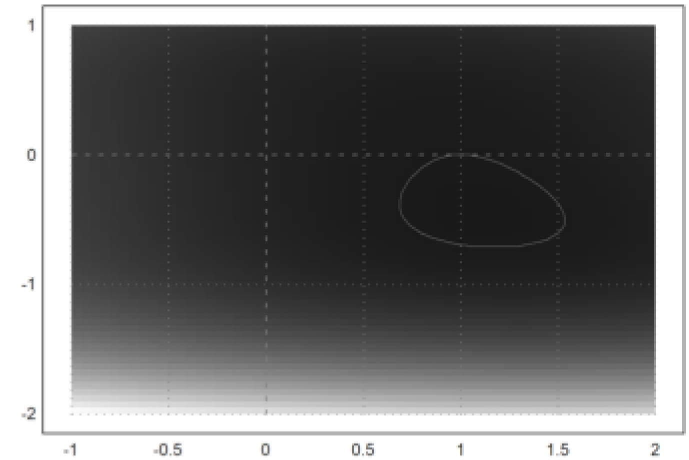
Gambar 127: images/PLOT%202D_Diva%20Nagita_23030630024-135.png



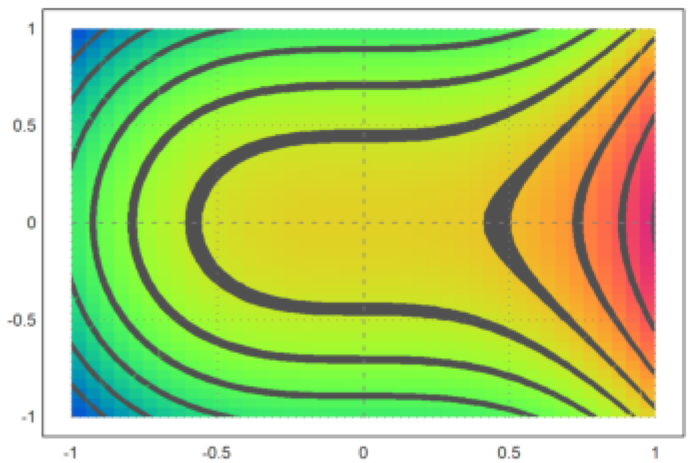
Gambar 128: images/PLOT%202D_Diva%20Nagita_23030630024-136.png



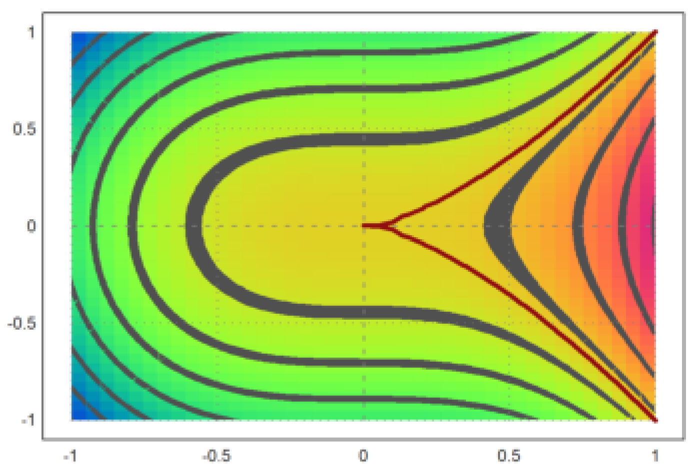
Gambar 129: images/PLOT%202D_Diva%20Nagita_23030630024-137.png



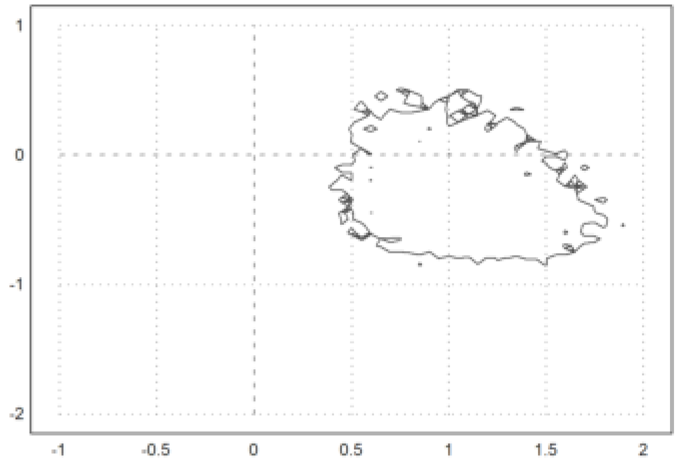
Gambar 130: images/PLOT%202D_Diva%20Nagita_23030630024-138.png



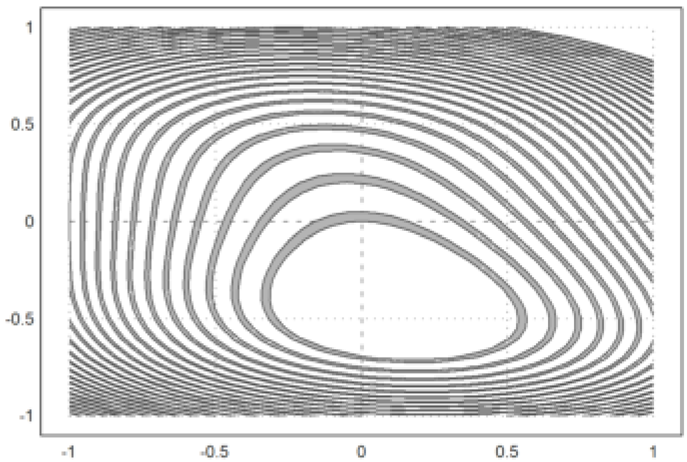
Gambar 131: images/PLOT%202D_Diva%20Nagita_23030630024-139.png



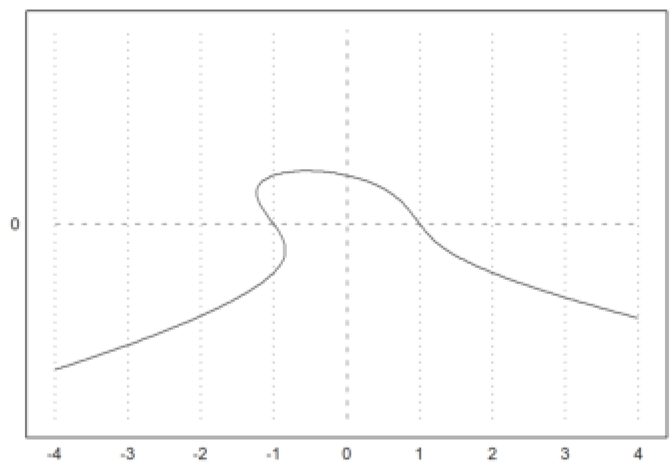
Gambar 132: images/PLOT%202D_Diva%20Nagita_23030630024-140.png



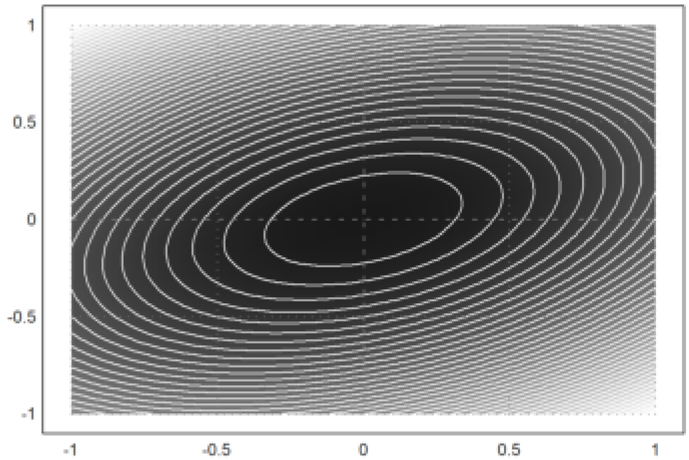
Gambar 133: images/PLOT%202D_Diva%20Nagita_23030630024-141.png



Gambar 134: images/PLOT%202D_Diva%20Nagita_23030630024-142.png



Gambar 135: images/PLOT%202D_Diva%20Nagita_23030630024-143.png



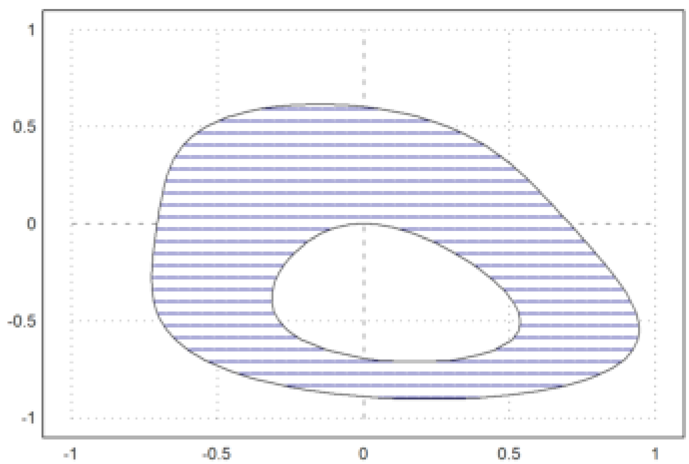
Gambar 136: images/PLOT%202D_Diva%20Nagita_23030630024-144.png

$$a \leq f(x, y) \leq b$$

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



Gambar 137: images/PLOT%202D_Diva%20Nagita_23030630024-146.png

Plot implisit juga dapat menunjukkan rentang level. Maka level harus berupa matriks 2xn interval level, di mana baris pertama berisi awal dan baris kedua adalah akhir dari setiap interval. Sebagai alternatif, vektor baris sederhana dapat digunakan untuk level, dan parameter dl memperluas nilai level ke interval.

>plot2d("x^4+y^4",r=1.5,level=[0;1],color=blue,style="/"):

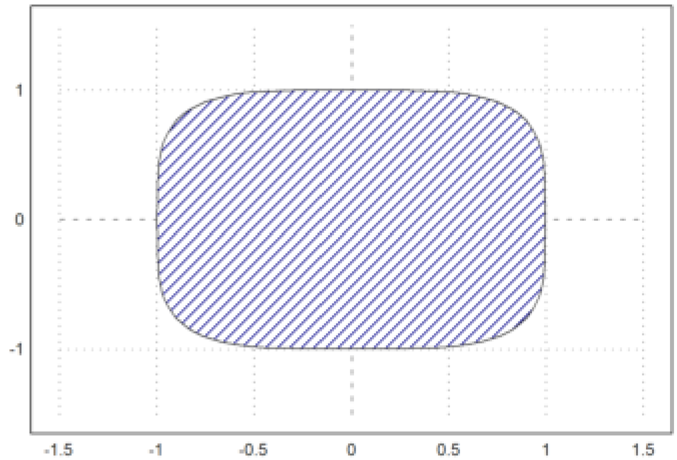
>plot2d("x^2+y^3+x*y",level=[0,2,4;1,3,5],style="/",r=2,n=100):

>plot2d("x^2+y^3+x*y",level=-10:20,r=2,style="-",dl=0.1,n=100):

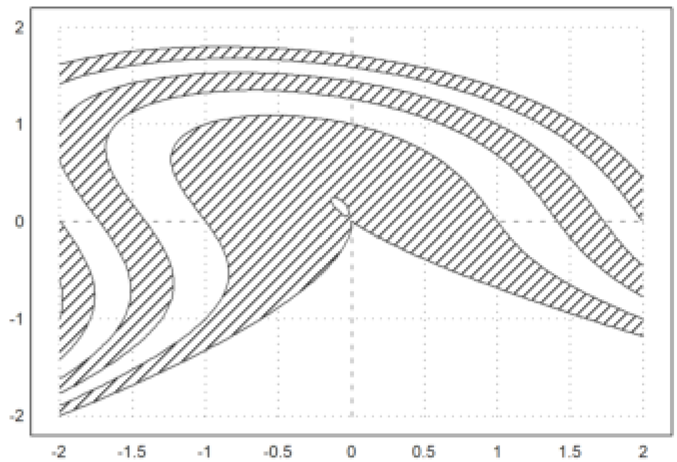
>plot2d("sin(x)*cos(y)",r=pi,>hue,>levels,n=100):

Anda juga dapat menandai suatu wilayah

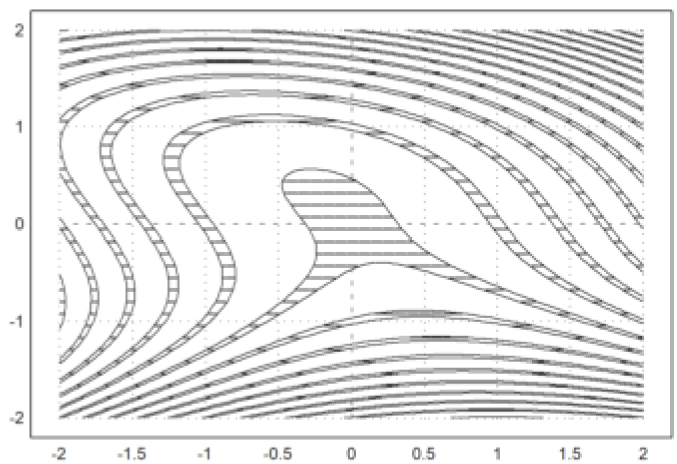
$$a \leq f(x, y) \leq b.$$



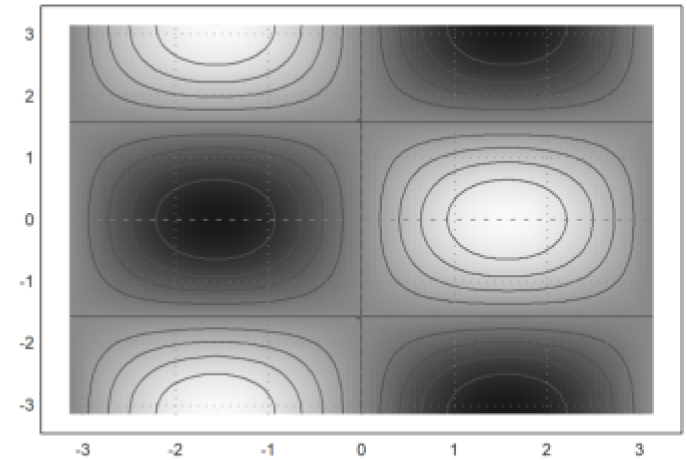
Gambar 138: images/PLOT%202D_Diva%20Nagita_23030630024-147.png



Gambar 139: images/PLOT%202D_Diva%20Nagita_23030630024-148.png



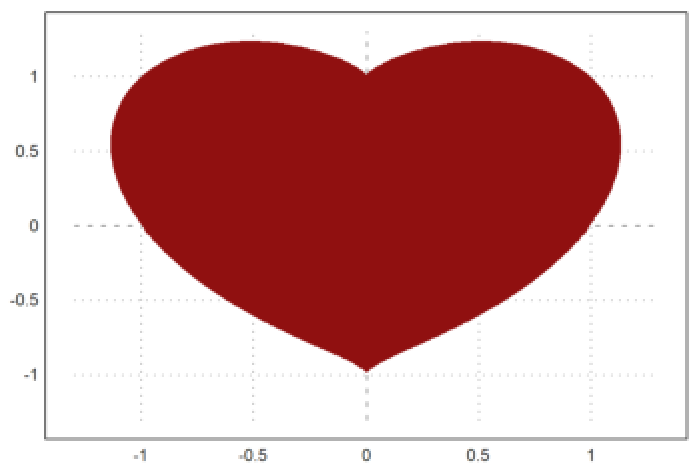
Gambar 140: images/PLOT%202D_Diva%20Nagita_23030630024-149.png



Gambar 141: images/PLOT%202D_Diva%20Nagita_23030630024-150.png

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):



Gambar 142: images/PLOT%202D_Diva%20Nagita_23030630024-152.png

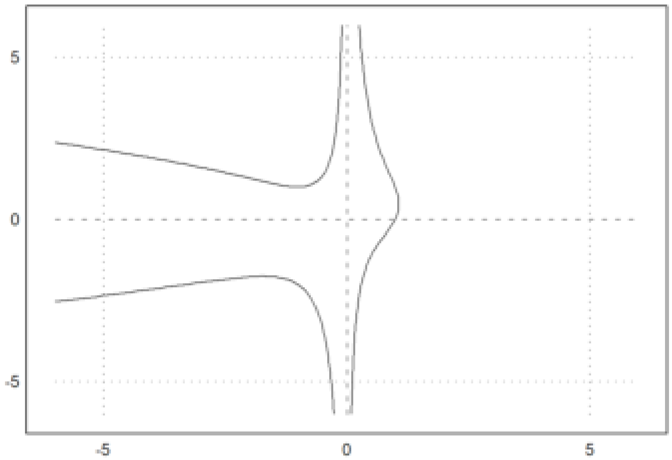
Dimungkinkan untuk menentukan level tertentu. Sebagai contoh, kita dapat memplot solusi dari persamaan seperti

$$x^3 - xy + x^2y^2 = 6$$

>plot2d(“x$^{3-x^*y+x}$2*y^2”,r=6,level=1,n=100):

>function starplot1 (v, style=“/”, color=green, lab=none) . . .

```
if !holding() then clg; endif;
w=window(); window(0,0,1024,1024);
h=holding(1);
r=max(abs(v))*1.2;
setplot(-r,r,-r,r);
n=cols(v); t=linspace(0,2pi,n);
```



Gambar 143: images/PLOT%202D_Diva%20Nagita_23030630024-154.png

```
  v=v|v[1]; c=v*cos(t); s=v*sin(t);
  cl=barcolor(color); st=barstyle(style);
  loop 1 to n
    polygon([0,c[#],c[#+1]],[0,s[#],s[#+1]],1);
    if lab!=none then
      rlab=v[#]+r*0.1;
      {col,row}=toscreen(cos(t[#])*rlab,sin(t[#])*rlab);
      ctext(""+lab[#],col,row-textheight()/2);
    endif;
  end;
  barcolor(cl); barstyle(st);
  holding(h);
  window(w);
endfunction
```

Tidak ada kisi-kisi atau kutu sumbu di sini. Selain itu, kami menggunakan jendela penuh untuk plot.

Kami memanggil reset sebelum kami menguji plot ini untuk mengembalikan default grafis. Hal ini tidak perlu dilakukan, jika Anda yakin bahwa plot Anda berfungsi.

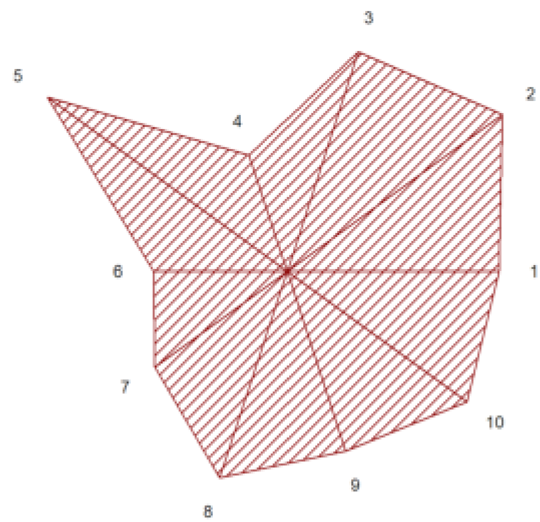
>reset; starplot1(normal(1,10)+5,color=red,lab=1:10):

Terkadang, Anda mungkin ingin memplot sesuatu yang tidak dapat dilakukan oleh plot2d, tetapi hampir.

Pada fungsi berikut ini, kita akan membuat plot impuls logaritmik. plot2d dapat membuat plot logaritmik, tetapi tidak untuk batang impuls.

>function logimpulseplot1 (x,y) ...

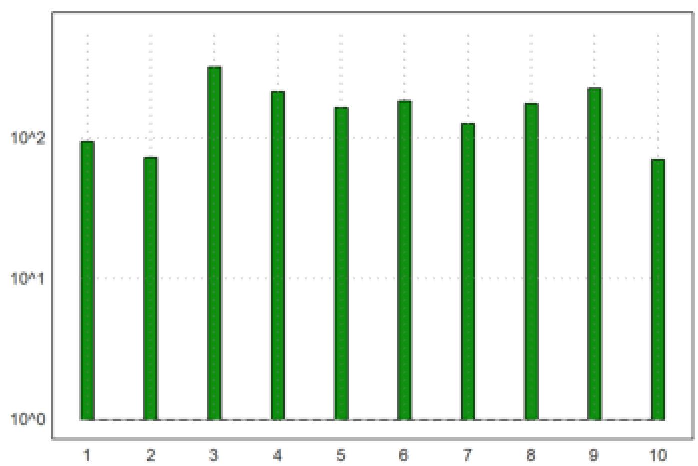
```
  {x0,y0}=makeimpulse(x,log(y)/log(10));
  plot2d(x0,y0,>bar,grid=0);
  h=holding(1);
  frame();
  xgrid(ticks(x));
  p=plot();
  for i=-10 to 10;
    if i<=p[4] and i>=p[3] then
       ygrid(i,yt="10^"+i);
    endif;
  end;
  holding(h);
endfunction
```



Gambar 144: images/PLOT%202D_Diva%20Nagita_23030630024-155.png

Mari kita uji dengan nilai yang terdistribusi secara eksponensial.

>aspect(1.5); x=1:10; y=-log(random(size(x)))*200; . . .
> logimpulseplot1(x,y):



Gambar 145: images/PLOT%202D_Diva%20Nagita_23030630024-156.png

Mari kita menghidupkan kurva 2D dengan menggunakan plot langsung. Perintah plot(x,y) hanya memplot kurva ke dalam jendela plot. setplot(a,b,c,d) mengatur jendela ini.

Fungsi wait(0) memaksa plot untuk muncul pada jendela grafik. Jika tidak, penggambaran 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 sembarang tombol untuk menghentikan animasi ini.

>animliss(2,3); // lihat hasilnya, jika sudah puas, tekan ENTER

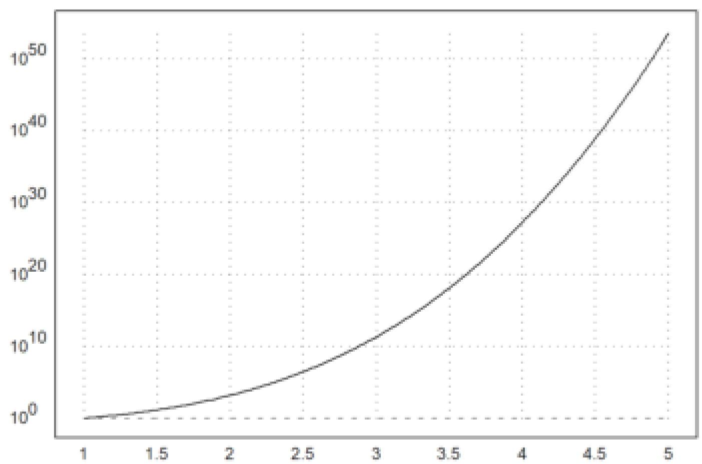
Plot Logaritmik

EMT menggunakan parameter “logplot” untuk skala logaritmik.

lot logaritmik dapat diplot menggunakan skala logaritmik dalam y dengan logplot = 1, atau menggunakan skala logaritmik dalam x dan y dengan logplot = 2, atau dalam x dengan logplot = 3.

- logplot=1: y-logarithmic
- logplot=2: x-y-logarithmic
- logplot=3: x-logarithmic

>plot2d(“exp(x$^{3-x)*x}$2”,1,5,logplot=1):



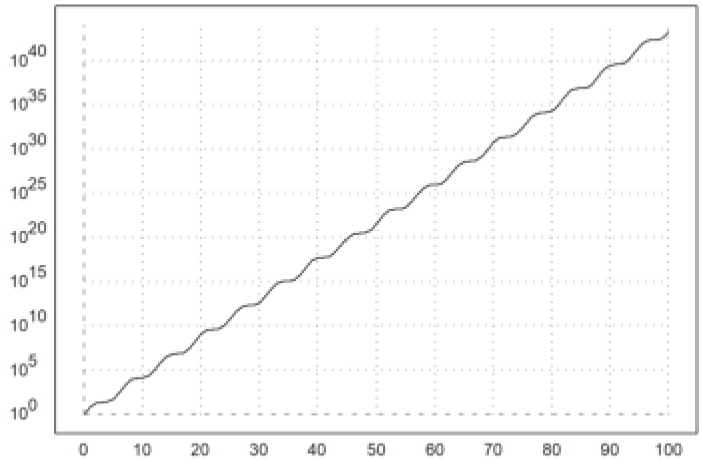
Gambar 146: images/PLOT%202D_Diva%20Nagita_23030630024-157.png

>plot2d(“exp(x+sin(x))”,0,100,logplot=1):

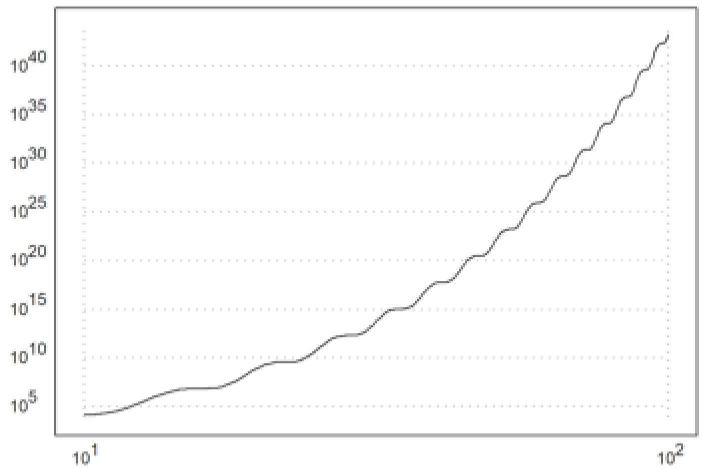
>plot2d(“exp(x+sin(x))”,10,100,logplot=2):

>plot2d(“gamma(x)”,1,10,logplot=1):

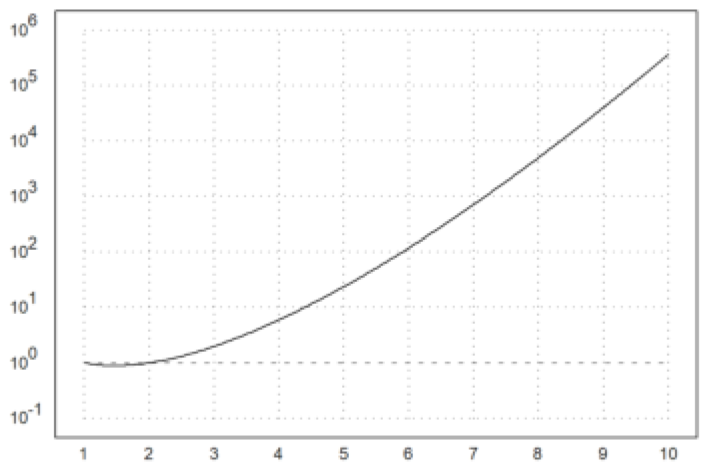
>plot2d(“log(x*(2+sin(x/100)))”,10,1000,logplot=3):



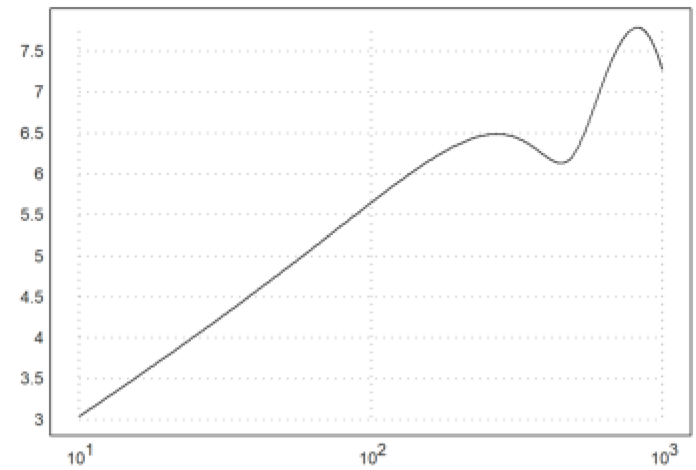
Gambar 147: images/PLOT%202D_Diva%20Nagita_23030630024-158.png



Gambar 148: images/PLOT%202D_Diva%20Nagita_23030630024-159.png

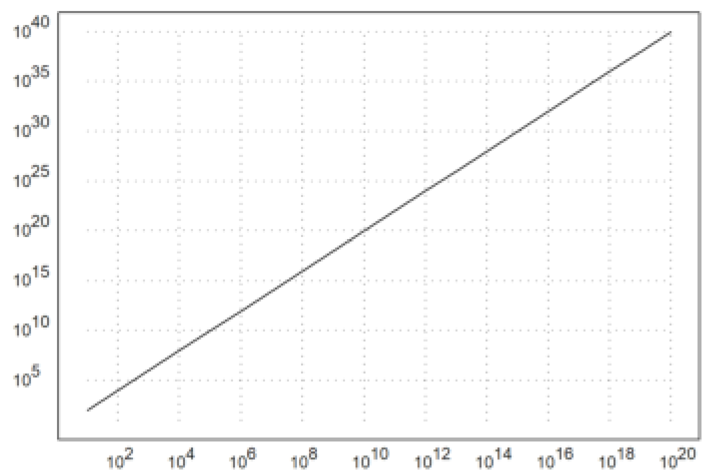


Gambar 149: images/PLOT%202D_Diva%20Nagita_23030630024-160.png



Gambar 150: images/PLOT%202D_Diva%20Nagita_23030630024-161.png

Hal ini juga bisa dilakukan dengan plot data.
>x=10ˆ(1:20); y=xˆ2-x;
>plot2d(x,y,logplot=2):



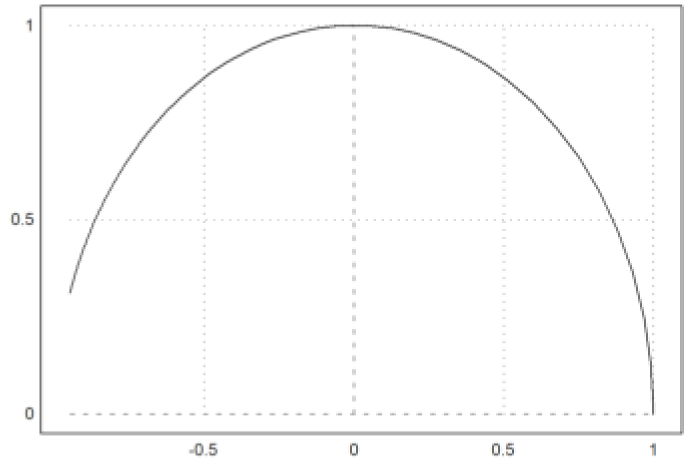
Gambar 151: images/PLOT%202D_Diva%20Nagita_23030630024-162.png

Latihan Soal

Buatlah grafik fungsi parametrik berikut:

$$x = \frac{1-t^2}{1+t^2}, y = \frac{2t}{1+t^2}, -1 \leq t \leq 1$$

>t=linspace(0,2*pi,100);. . .
> plot2d((1-t$^{2)/(1+t}$2),(2t)/(1+tˆ2)):
Grafik tersebut akan menampilkan lingkaran dengan pusat (0,0) dan jari-jari 1.
Dapat diperhatikan bahwa kurva hanya mencakup setengah lingkaran karna rentang nilai t dibatasi dari -1 sampai 1.
Buatlah grafik fungsi parametrik berikut:

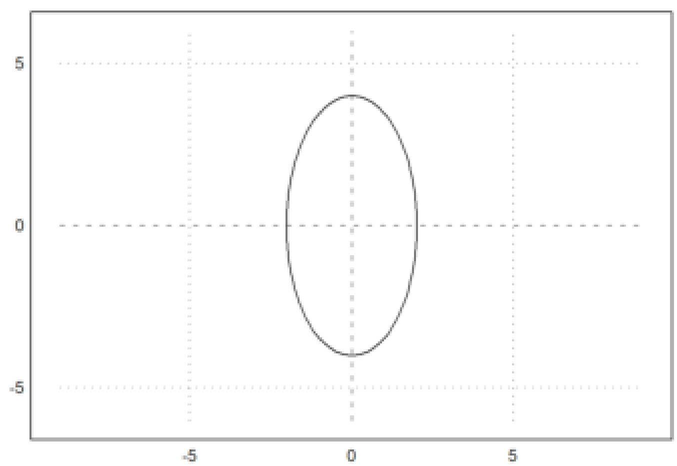


Gambar 152: images/PLOT%202D_Diva%20Nagita_23030630024-164.png

$$x = 2.cos(t)$$

$$y = 4.sin(t)$$

>t=linspace(0,2*pi,100);. . .
> plot2d(2*cos(t),4*sin(t),r=6):



Gambar 153: images/PLOT%202D_Diva%20Nagita_23030630024-167.png

Kode diatas akan menghasilkan Grafik Fungsi Parametrik yang membentuk lingkaran dengan jari-jari 2 pada sumbu x dan jari-jari 4 pada sumbu y

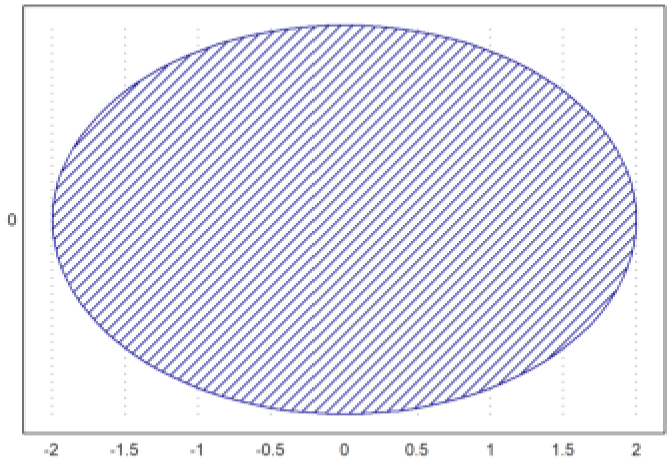
Penambahan r=5 untuk melihat bahwa grafik yang terbentuk bukan lingkaran tetapi elips.

Buatlah grafik fungsi parametrik diatas dengan bidang yang memenuhi syarat menjadi terarsir

>t=linspace(0,2*pi,100); x=2*cos(t); y=4*sin(t);. . .
> plot2d(x,y,>filled,fillcolor=blue,style="/"):

Rujukan Lengkap Fungsi plot2d()

function plot2d (xv, yv, btest, a, b, c, d, xmin, xmax, r, n, ..
logplot, grid, frame, framecolor, square, color, thickness, style, ..



Gambar 154: images/PLOT%202D_Diva%20Nagita_23030630024-168.png

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

"[]", "&lt;&gt;", ".", "..", "...",

"*", "+", "|", "-", "o"

"[]#", "&lt;&gt;#", "o#" (filled shapes)

"[]w", "&lt;&gt;w", "ow" (non-transparent)

For lines use

"-", "--", "-.", ".", ".-.", "-.-", "-&gt;"

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 [-2,2] should be used. The y-range will be computed automatically, unless c and d are specified, or a radius r, which yields the range [-r,r] for x and y. For plots of functions, plot2d will use an adaptive evaluation of the function by default. To speed up the plot for complicated functions, switch this off with <adaptive, and optionally decrease the number of intervals n. Moreover, plot2d() will by default use mapping. I.e., it will compute the plot element for element. If your expression or your functions can handle a vector x, you can switch that off with <maps for faster evaluation. Note that adaptive plots are always computed element for element. If functions or expressions for both xv and for yv are specified, plot2d() will compute a curve with the xv values as x-coordinates and the yv values as y-coordinates. In this case, a range should be defined for the parameter using xmin, xmax. Expressions contained in strings must always be expressions in the parameter variable x.

unicodehyperref hyphensurl [ ]book xcolor amsmath,amssymb iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype [protrusion]basicmath parskip graphicx bookmark xurl same hidelinks, pdfcreator=LaTeX via pandoc

MENGGAMBAR PLOT 3D DENGAN EMT

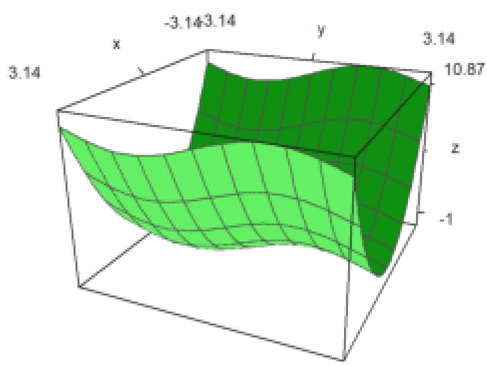
Ini adalah pengenalan plot 3D di Euler. Kita membutuhkan plot 3D untuk memvisualisasikan fungsi dari dua variabel.

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):



Gambar 155: images/PLOT3D_Diva%20Nagita_23030630024-001.png

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.

Grafik Fungsi Linear

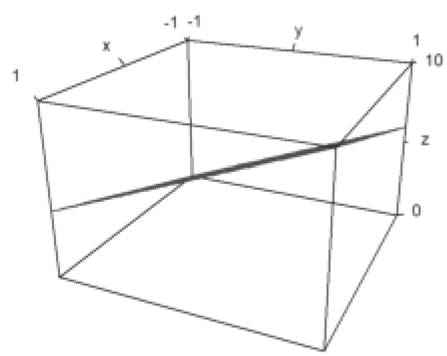
Fungsi linear dua variabel biasanya dinyatakan dalam bentuk

$$f(x, y) = ax + by + c$$

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"):



Gambar 156: images/PLOT3D_Diva%20Nagita_23030630024-005.png

$$f(x, y) = -2x - 3y - 5$$

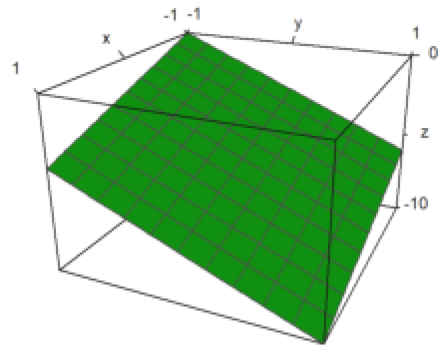
>plot3d("-2*x-3*y-5"):

$$f(x, y) = 5x - 7y + 9$$

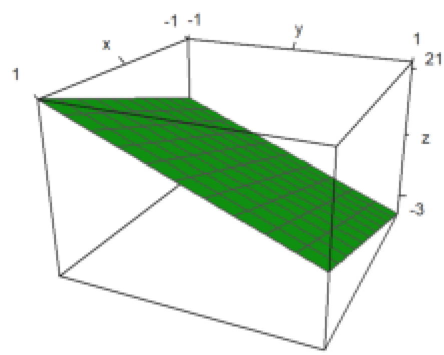
>plot3d("5*x-7*y+9"):

Grafik Fungsi Kuadrat

Fungsi kuadrat dua variabel biasanya dinyatakan dalam bentuk



Gambar 157: images/PLOT3D_Diva%20Nagita_23030630024-007.png



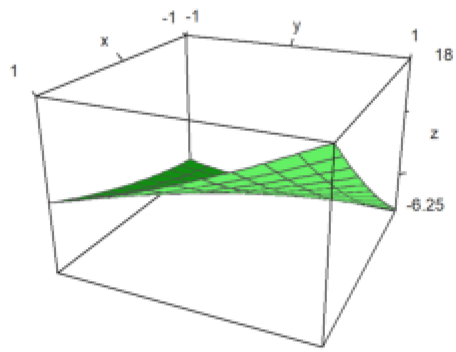
Gambar 158: images/PLOT3D_Diva%20Nagita_23030630024-009.png

$$f(x, y) = ax^2 + by^2 + cxy + dx + ey + f$$

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:

$$f(x, y) = x^2 + y^2 + 4xy + 8x + 3y + 1$$

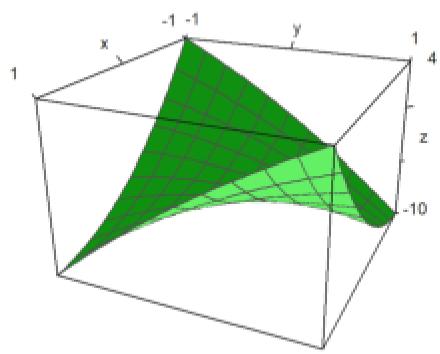
>plot3d(“x^{2+y}2+4*x*y+8*x+3*y+1”):



Gambar 159: images/PLOT3D_Diva%20Nagita_23030630024-012.png

$$f(x, y) = 3x^2 - y^2 + 7xy - 5$$

>plot3d(“3*x^{2-y}2+7*x*y-5”):



Gambar 160: images/PLOT3D_Diva%20Nagita_23030630024-014.png

Grafik Fungsi Akar Kuadrat

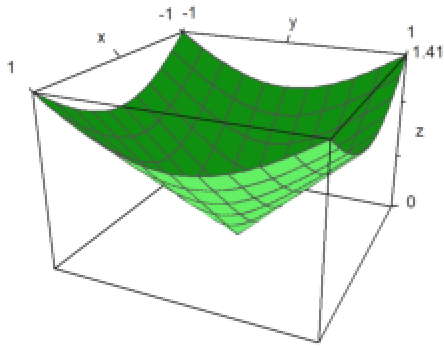
Grafik fungsi akar kuadrat dua variabel

$$f(x,y) = \sqrt{x^2 + y^2}$$

adalah grafik permukaan yang menggambarkan jarak titik (x, y) dari titik asal (0, 0) dalam ruang tiga dimensi. Contoh:

$$f(x,y) = \sqrt{x^2 + y^2}$$

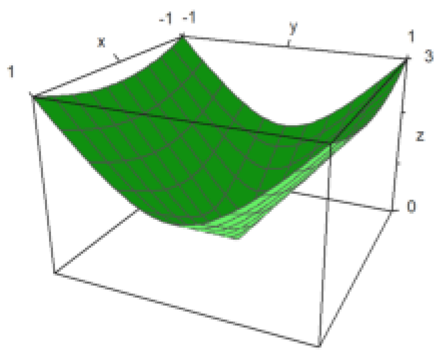
>plot3d(“sqrt(x^{2+y}2)”):



Gambar 161: images/PLOT3D_Diva%20Nagita_23030630024-017.png

$$f(x,y) = \sqrt{2x^2 + 7y^2}$$

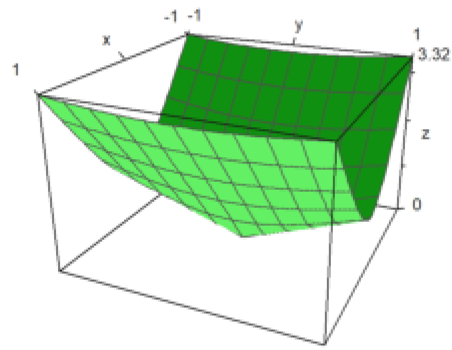
>plot3d(“sqrt(2*x^{2+7*y}2)”):



Gambar 162: images/PLOT3D_Diva%20Nagita_23030630024-019.png

$$f(x,y) = \sqrt{10x^2 + y^2}$$

>plot3d(“sqrt(10*x^{2+y}2)”):



Gambar 163: images/PLOT3D_Diva%20Nagita_23030630024-021.png

Grafik Fungsi Eksponensial

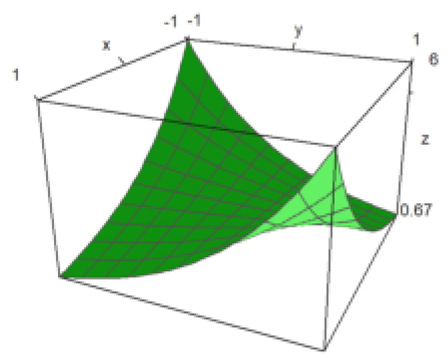
Fungsi eksponensial dua variabel bisa dinyatakan

$$f(x, y) = a.b^{xy}$$

dimana a dan b adalah konstanta, x dan y adalah variabel. Fungsi ini menggambarkan pertumbuhan eksponensial yang bergantung pada nilai x dan y.
Contoh:

$$f(x, y) = 2.3^{xy}$$

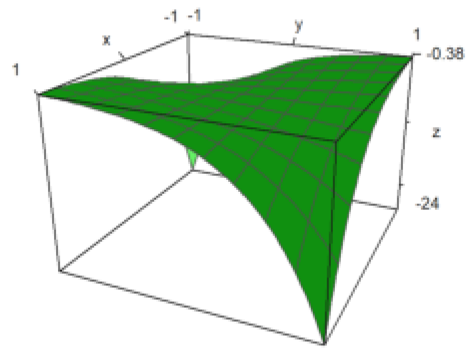
>plot3d("2*3^(x*y)"):



Gambar 164: images/PLOT3D_Diva%20Nagita_23030630024-024.png

$$f(x, y) = -3.8^{xy}$$

>plot3d("-3*8^(x*y)"):



Gambar 165: images/PLOT3D_Diva%20Nagita_23030630024-026.png

Grafik Fungsi Logaritma

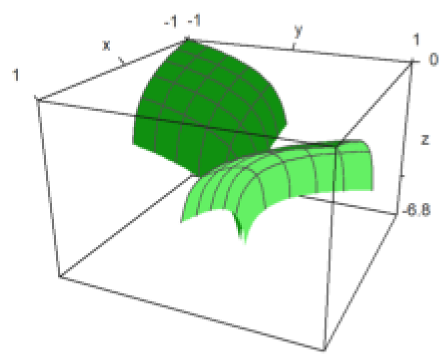
Grafik fungsi logaritma dua variabel adalah grafik yang menggambarkan nilai logaritma dari suatu ekspresi yang melibatkan dua variabel (biasanya x dan y). Fungsi logaritma dua variabel ini dinyatakan sebagai

$$f(x,y) = log_b(xy)$$

dimana b adalah basis logaritma. Basis logaritma ini dapat berbeda-beda.
Contoh:

$$f(x,y) = log(xy), basis 10$$

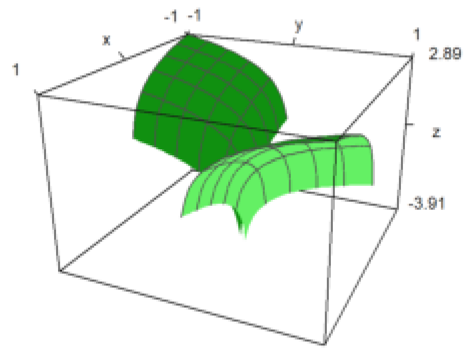
>plot3d("log(x*y)"):



Gambar 166: images/PLOT3D_Diva%20Nagita_23030630024-029.png

$$f(x,y) = log(2x.9y), basis 10$$

>plot3d("log(2x*9y)"):



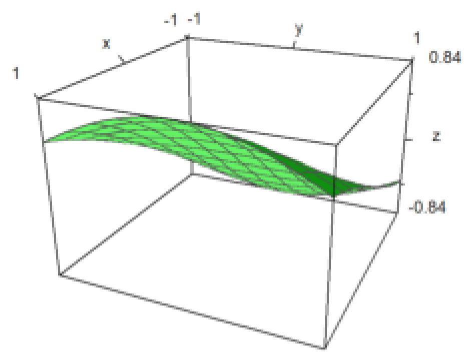
Gambar 167: images/PLOT3D_Diva%20Nagita_23030630024-031.png

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)"):



Gambar 168: images/PLOT3D_Diva%20Nagita_23030630024-033.png

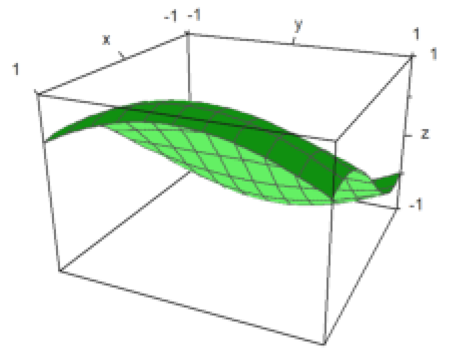
$$f(x,y) = sin(2x).cos(y)$$

>plot3d("sin(2x)*cos(y)"):

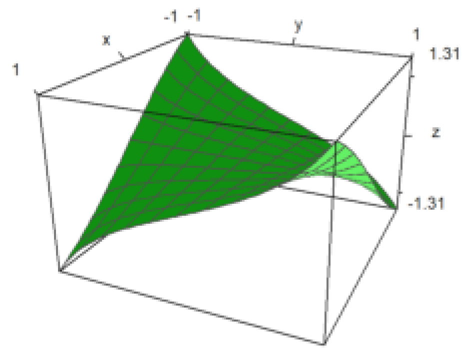
$$f(x,y) = sin(x).tan(y)$$

>plot3d("sin(x)*tan(y)"):

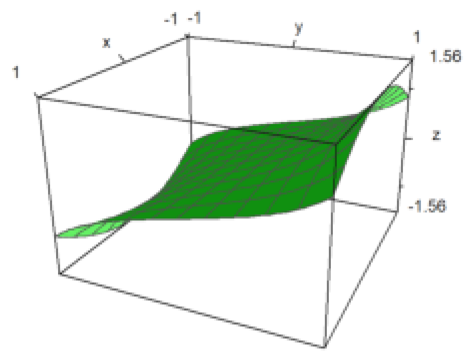
$$f(x,y) = cos(x).tan(y)$$



Gambar 169: images/PLOT3D_Diva%20Nagita_23030630024-035.png



Gambar 170: images/PLOT3D_Diva%20Nagita_23030630024-037.png

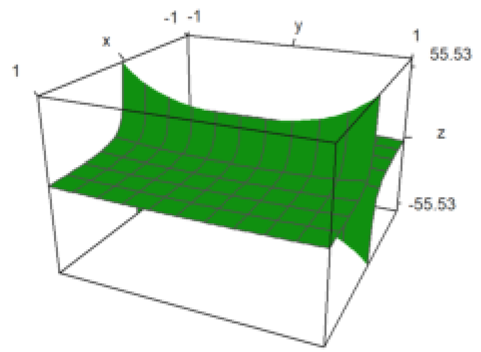


Gambar 171: images/PLOT3D_Diva%20Nagita_23030630024-039.png

>plot3d("cos(x)*tan(y)"):

$$f(x,y) = cosec(x).sec(y)$$

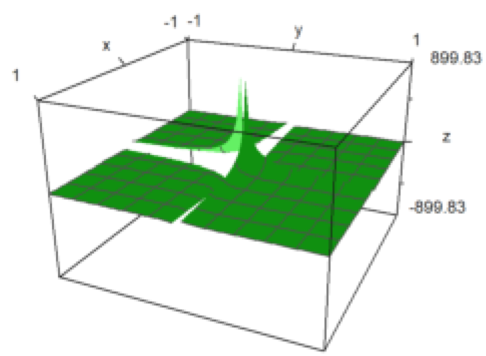
>plot3d("cosec(x)*sec(y)"):



Gambar 172: images/PLOT3D_Diva%20Nagita_23030630024-041.png

$$f(x,y) = cot(x).cosec(y)$$

>plot3d("cot(x)*cosec(y)"):



Gambar 173: images/PLOT3D_Diva%20Nagita_23030630024-043.png

Grafik Fungsi Nilai Mutlak

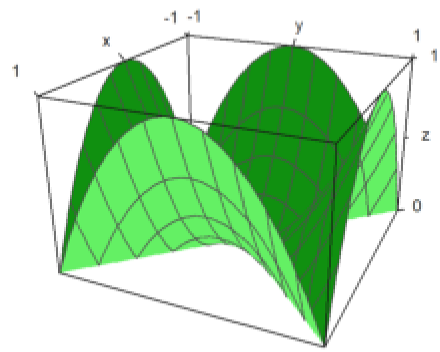
Fungsi nilai mutlak dua variabel, juga dikenal sebagai fungsi modul dua variabel, dinyatakan sebagai

$$f(x,y) = |g(x,y)|$$

dimana g(x,y) adalah fungsi dua variabel.
Contoh:

$$f(x,y) = |x^2 - y^2|$$

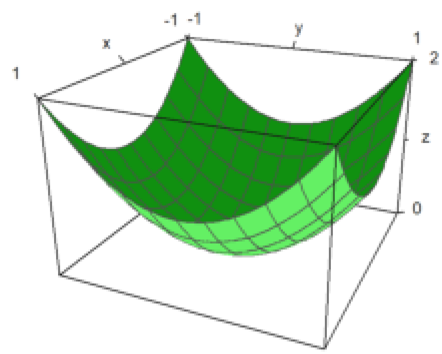
>plot3d("abs(x^2 - y^2)"):



Gambar 174: images/PLOT3D_Diva%20Nagita_23030630024-046.png

$$f(x,y) = |x^2 + y^2|$$

>plot3d("abs(x^2 + y^2)"):



Gambar 175: images/PLOT3D_Diva%20Nagita_23030630024-048.png

$$f(x,y) = |-2x^2 - 5y^2|$$

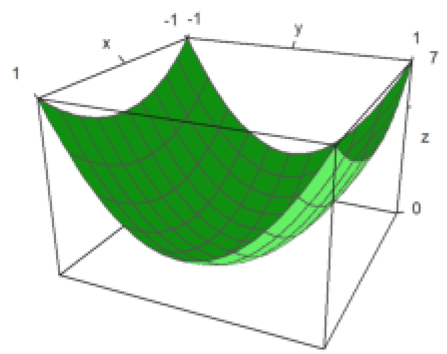
>plot3d("abs(-2x^2 - 5y^2)"):

$$f(x,y) = |x^2 - 8y^2|$$

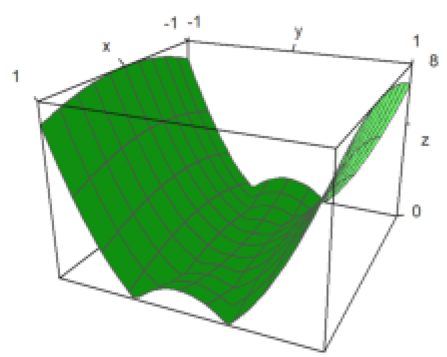
>plot3d("abs(x^2 - 8y^2)"):

Latihan soal

Buatkan grafik dari fungsi berikut:



Gambar 176: images/PLOT3D_Diva%20Nagita_23030630024-050.png

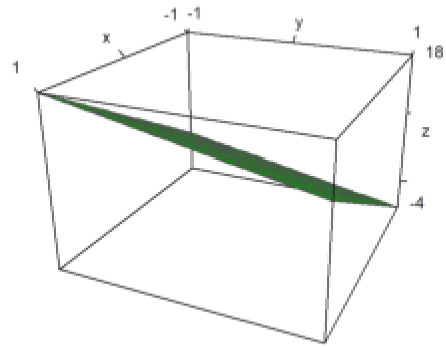


Gambar 177: images/PLOT3D_Diva%20Nagita_23030630024-052.png

1.

$$f(x,y) = 8x - 3y + 7$$

>plot3d("8*x - 3*y +7"):

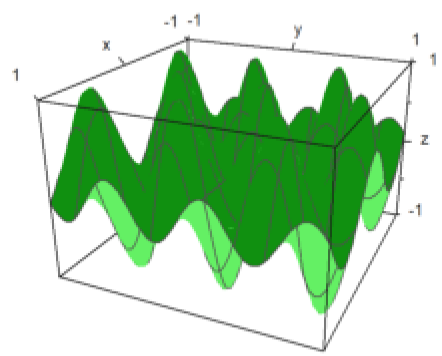


Gambar 178: images/PLOT3D_Diva%20Nagita_23030630024-054.png

2.

$$f(x,y) = cos(5x).sin(9y)$$

>plot3d("cos(5*x)*sin(9*y)"):

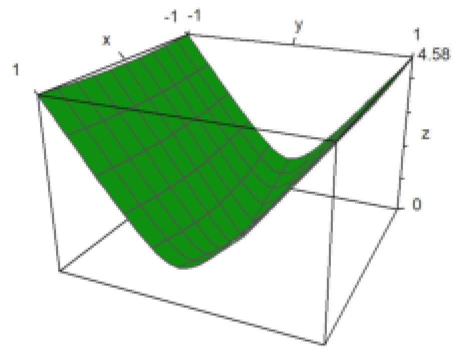


Gambar 179: images/PLOT3D_Diva%20Nagita_23030630024-056.png

3.

$$f(x,y) = \sqrt{x^2 + 20y^2}$$

>plot3d("sqrt(x^{2+20*y}2)"):



Gambar 180: images/PLOT3D_Diva%20Nagita_23030630024-058.png

Menggambar Grafik Fungsi Dua Variabel yang Rumusnya Disimpan dalam Variabel Ekspresi

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

2. 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) = ax^2 + by^2 + 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

f(x, y) = sin(x) + cos(y)

akan menghasilkan permukaan yang berulang-ulang naik dan turun.

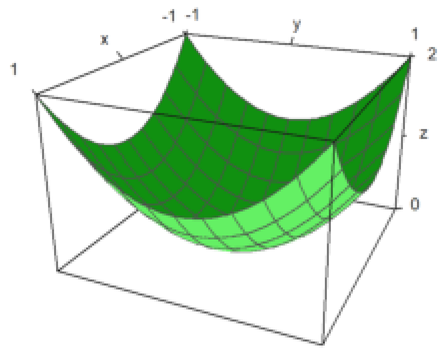
4.Fungsi Pecahan

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.

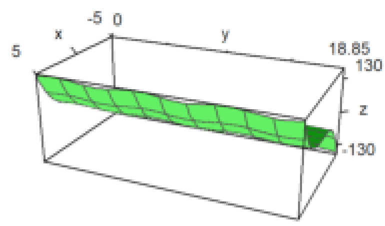
Menggambar Grafik Fungsi

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(“$x^{2+y}2$”):



Gambar 181: images/PLOT3D_Diva%20Nagita_23030630024-059.png

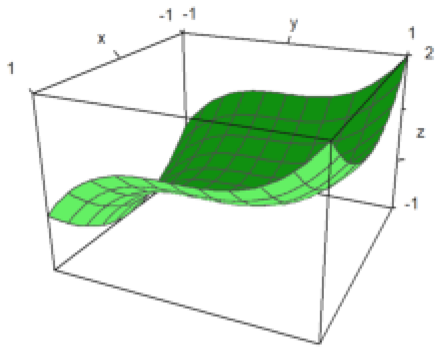
>plot3d(“xˆ3+x*sin(y)”,-5,5,0,6*pi):



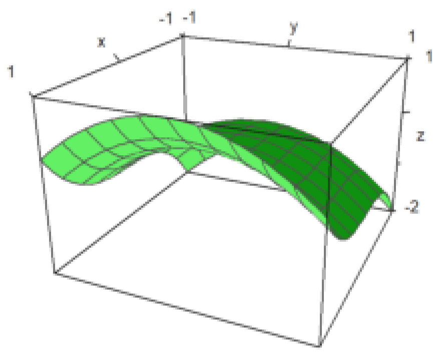
Gambar 182: images/PLOT3D_Diva%20Nagita_23030630024-060.png

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”):



Gambar 183: images/PLOT3D_Diva%20Nagita_23030630024-061.png



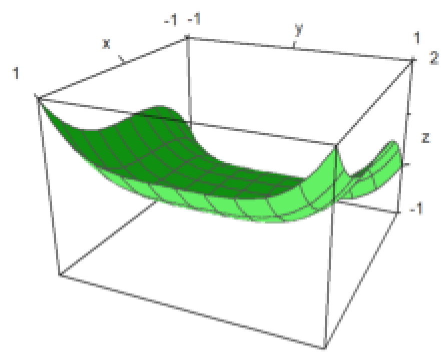
Gambar 184: images/PLOT3D_Diva%20Nagita_23030630024-062.png

Contoh Latihan Soal

$$f(x,y) = x^3 + y^4$$

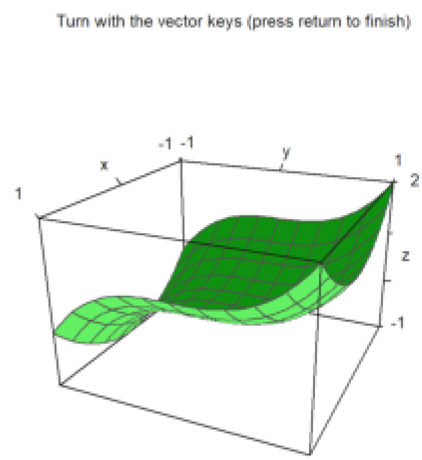
>function f(x,y):= x^{3+y}4

>plot3d("f"):



Gambar 185: images/PLOT3D_Diva%20Nagita_23030630024-064.png

>plot3d("a",>user, . . .
> title="Turn with the vector keys (press return to finish)"):



Gambar 186: images/PLOT3D_Diva%20Nagita_23030630024-065.png

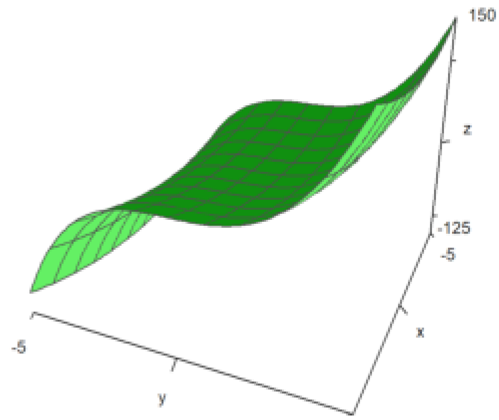
Perintah ini mengizinkan pengguna untuk menggambar grafik 3D dari fungsi yang mereka masukkan sendiri, serta memberikan petunjuk interaktif tentang cara berinteraksi dengan grafik. Pengguna dapat memutar tampilan grafik menggunakan tombol arah pada keyboard, dan menekan "return" untuk menyelesaikan interaksi.

>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.

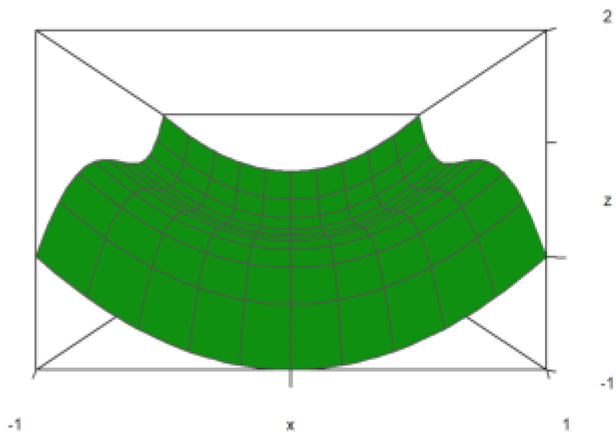
>view

```
[5,   2.6,   2,   0.4]
```



Gambar 187: images/PLOT3D_Diva%20Nagita_23030630024-066.png

>plot3d("a",distance=3,zoom=2,angle=0,height=0):



Gambar 188: images/PLOT3D_Diva%20Nagita_23030630024-067.png

>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):

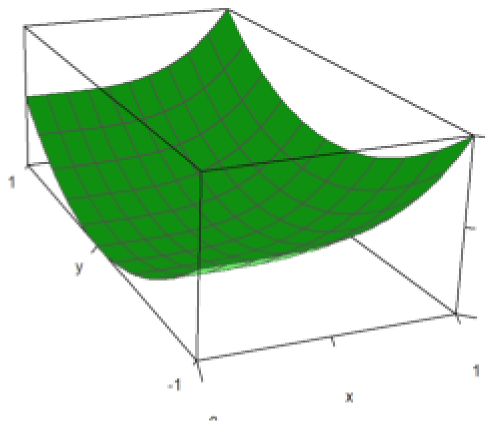
FUNGSI LINEAR

>function e(x,y):= 20x+10y-5

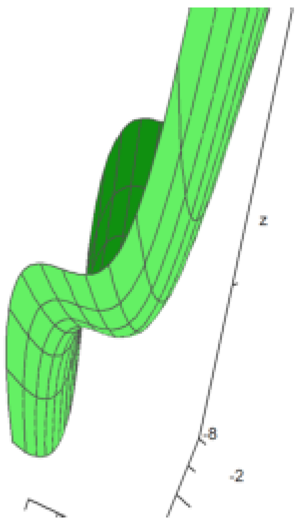
>plot3d("e"):

>plot3d("e",>user, . . .
> title="Turn with the vector keys (press return to finish)"):

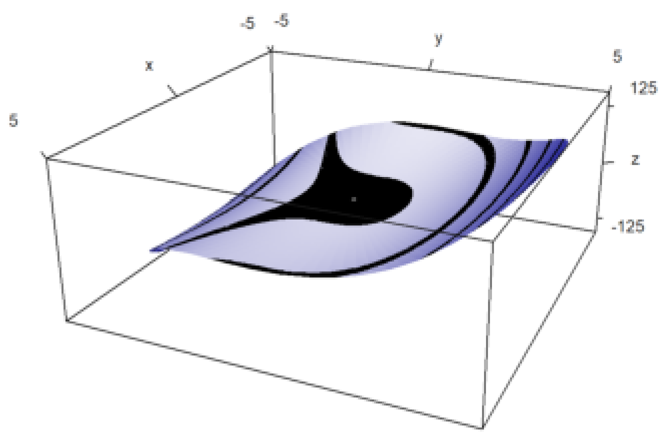
>plot3d("e",r=10,n=80,fscale=4,scale=1.2,frame=3):



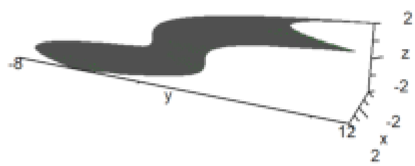
Gambar 189: images/PLOT3D_Diva%20Nagita_23030630024-068.png



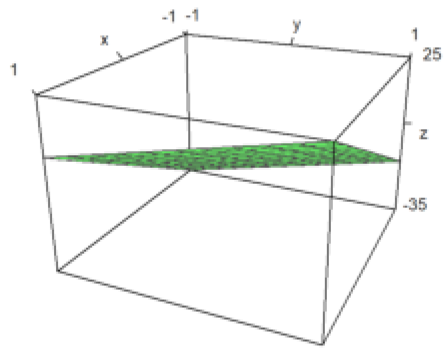
Gambar 190: images/PLOT3D_Diva%20Nagita_23030630024-069.png



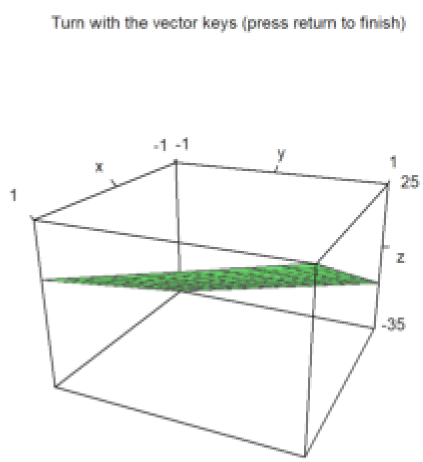
Gambar 191: images/PLOT3D_Diva%20Nagita_23030630024-070.png



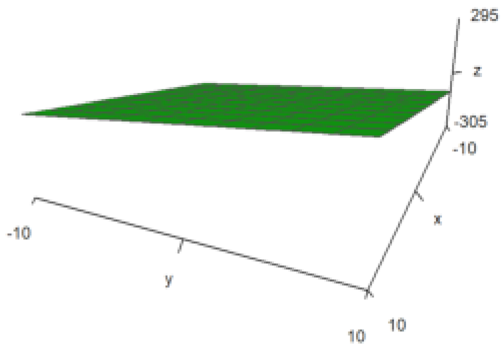
Gambar 192: images/PLOT3D_Diva%20Nagita_23030630024-071.png



Gambar 193: images/PLOT3D_Diva%20Nagita_23030630024-072.png



Gambar 194: images/PLOT3D_Diva%20Nagita_23030630024-073.png

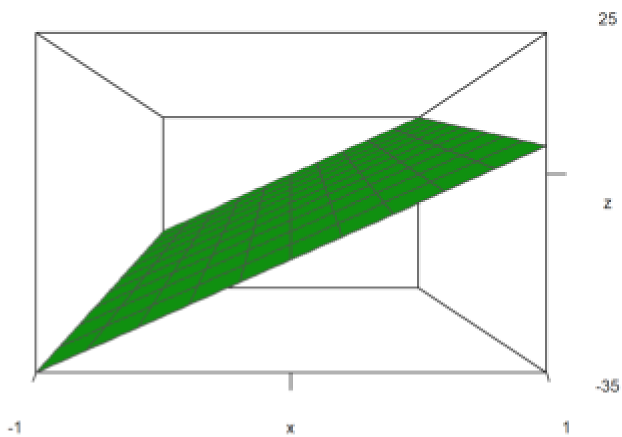


Gambar 195: images/PLOT3D_Diva%20Nagita_23030630024-074.png

>view

```
[5,  2.6,  2,  0.4]
```

>plot3d("e",distance=3,zoom=2,angle=0,height=0):



Gambar 196: images/PLOT3D_Diva%20Nagita_23030630024-075.png

>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):

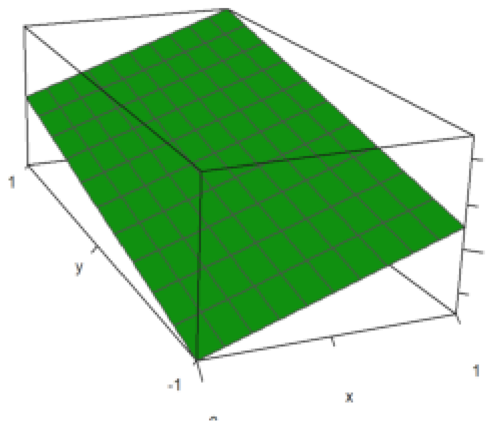
>plot3d("e",r=5,>polar, ...
> fscale=2,>hue,n=100,zoom=4,>contour,color=gray):

>plot3d("x","e","y",r=2,zoom=3.5,frame=3):

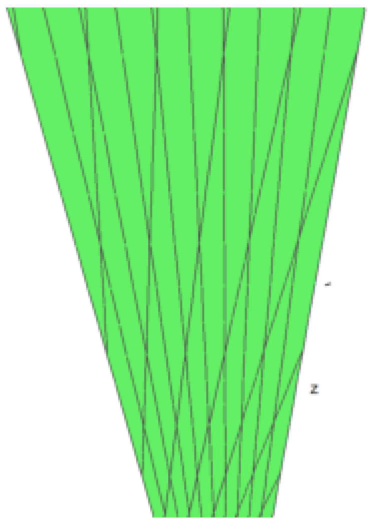
FUNGSI TRIGONOMETRI

>function f(x,y):= sin(x+y)

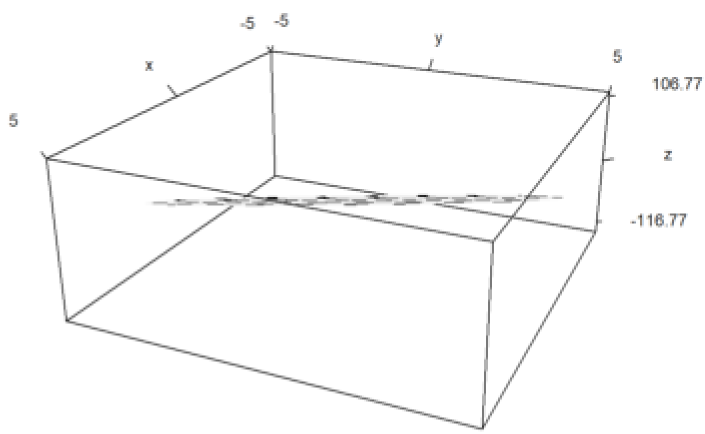
>plot3d("f")



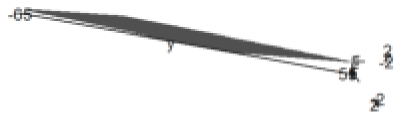
Gambar 197: images/PLOT3D_Diva%20Nagita_23030630024-076.png



Gambar 198: images/PLOT3D_Diva%20Nagita_23030630024-077.png

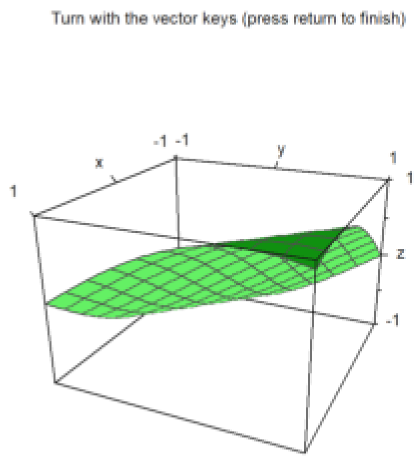


Gambar 199: images/PLOT3D_Diva%20Nagita_23030630024-078.png



Gambar 200: images/PLOT3D_Diva%20Nagita_23030630024-079.png

>plot3d("f",>user, . . .
> title="Turn with the vector keys (press return to finish)"):

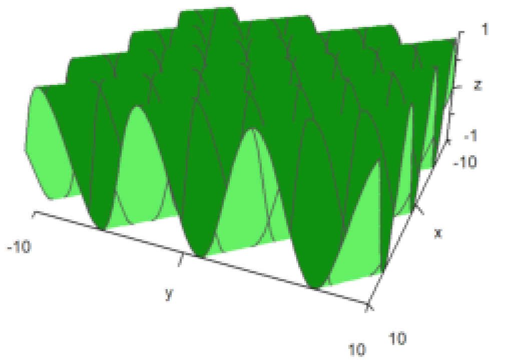


Gambar 201: images/PLOT3D_Diva%20Nagita_23030630024-080.png

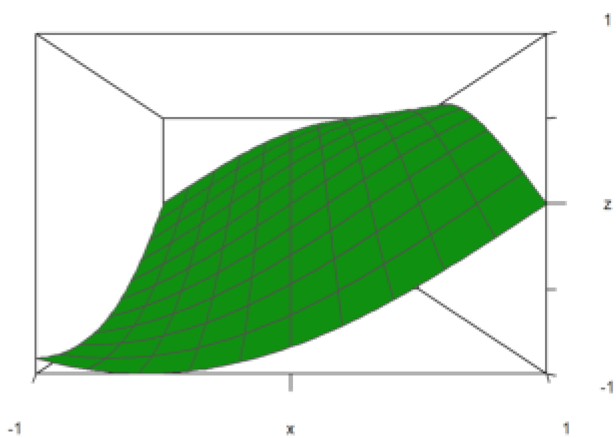
>plot3d("f",r=10,n=80,fscale=4,scale=1.2,frame=3):
>view

```
[5,  2.6,  2,  0.4]
```

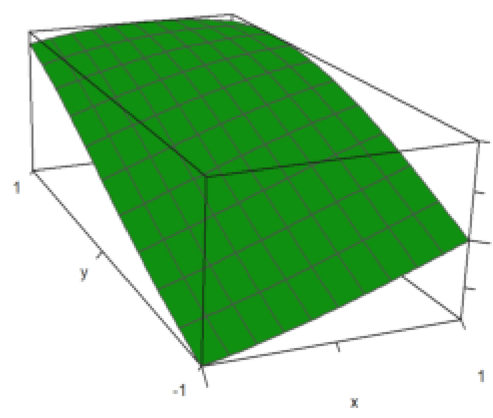
>plot3d("f",distance=3,zoom=2,angle=0,height=0):
>plot3d("f",a=0,b=1,c=-1,d=1,angle=-20°,height=20°, . . .
> center=[0.4,0,0],zoom=5):
>plot3d("f",r=5,>polar, . . .
> fscale=2,>hue,n=100,zoom=4,>contour,color=gray):



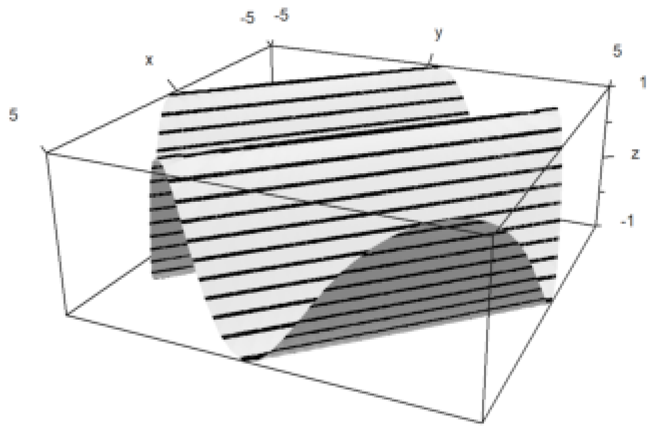
Gambar 202: images/PLOT3D_Diva%20Nagita_23030630024-081.png



Gambar 203: images/PLOT3D_Diva%20Nagita_23030630024-082.png



Gambar 204: images/PLOT3D_Diva%20Nagita_23030630024-083.png



Gambar 205: images/PLOT3D_Diva%20Nagita_23030630024-084.png

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

$$2(1) + 1 = 3$$

Visualisasi Grafik
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)

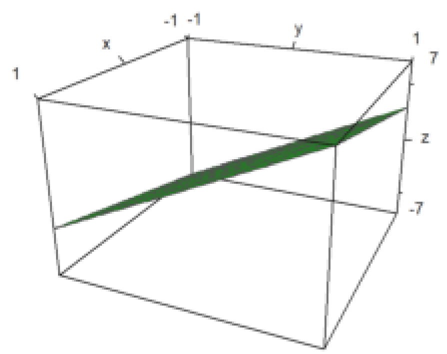
2) sintaks plot3d

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"):



Gambar 206: images/PLOT3D_Diva%20Nagita_23030630024-088.png

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

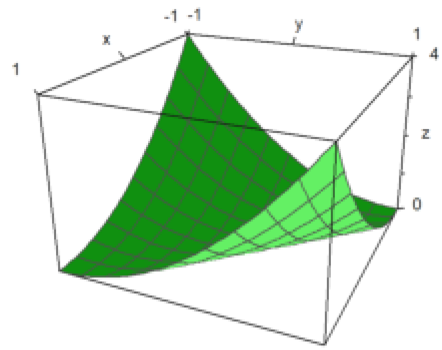
$$g(x, y) = x^{2y+8}$$

>function g(x,y):= x^(2y+8)
>plot3d("g"):

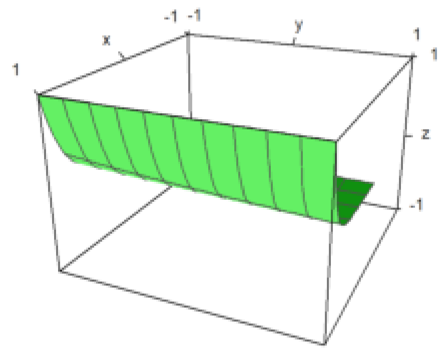
4. Grafik Fungsi Logaritma Dua Variabel

$$f(x, y) = \log(xy)$$

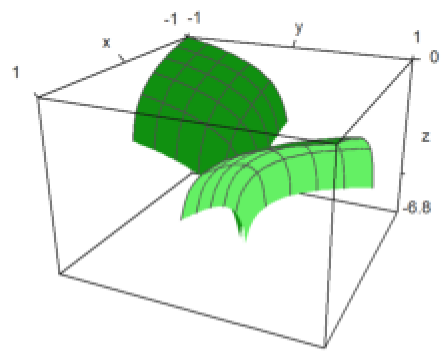
>function f(x,y):= log(x*y)
>plot3d("f"):



Gambar 207: images/PLOT3D_Diva%20Nagita_23030630024-090.png



Gambar 208: images/PLOT3D_Diva%20Nagita_23030630024-092.png

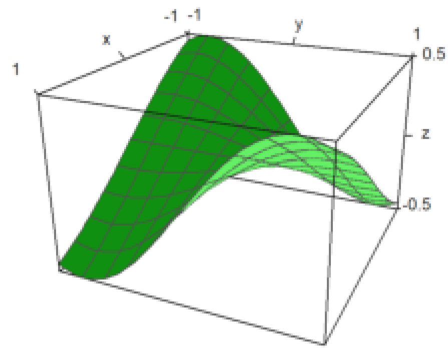


Gambar 209: images/PLOT3D_Diva%20Nagita_23030630024-094.png

5. Grafik Fungsi Trigonometri Dua Variabel

$$h(x, y) = sin(xy)cos(y)$$

>function h(x,y):= sin(x*y)*cos(y)
>plot3d("h"):

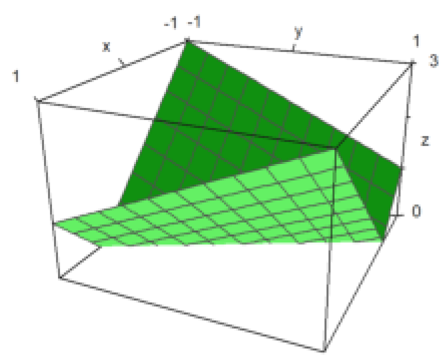


Gambar 210: images/PLOT3D_Diva%20Nagita_23030630024-096.png

6. Grafik Fungsi Nilai Mutlak Dua Variabel

$$i(x, y) = |(2x + y)|$$

>function i(x,y):= abs(2x+y)
>plot3d("i"):



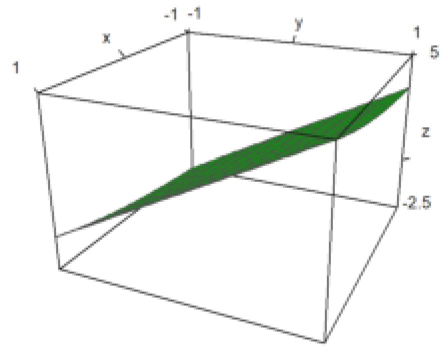
Gambar 211: images/PLOT3D_Diva%20Nagita_23030630024-098.png

Latihan Soal

Buatlah visualisasi grafik dari fungsi berikut ini!

$$f(x, y) = 2^x + 3y$$

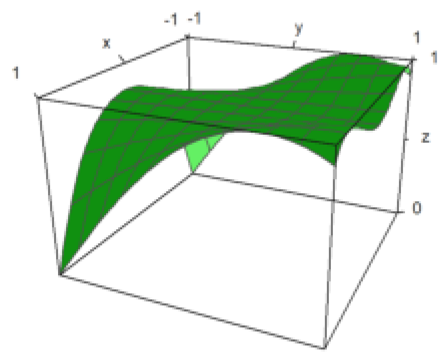
>function f(x,y):=2^x+3*y
>plot3d ("f"):



Gambar 212: images/PLOT3D_Diva%20Nagita_23030630024-100.png

$$g(x, y) = sin(x^2y + 1)$$

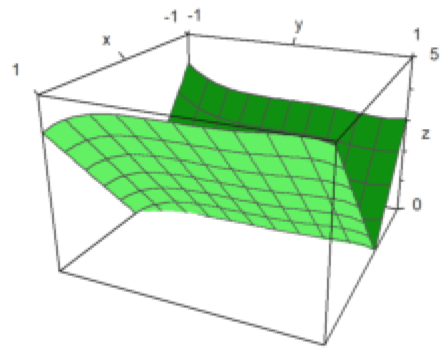
>function g(x,y):= sin(x^2*y+1)
>plot3d("g"):



Gambar 213: images/PLOT3D_Diva%20Nagita_23030630024-102.png

$$h(x, y) = |4x + sin(y^3 + 1)|$$

>function h(x,y):=abs(4*x + sin(y^3+1))
>plot3d("h"):



Gambar 214: images/PLOT3D_Diva%20Nagita_23030630024-104.png

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

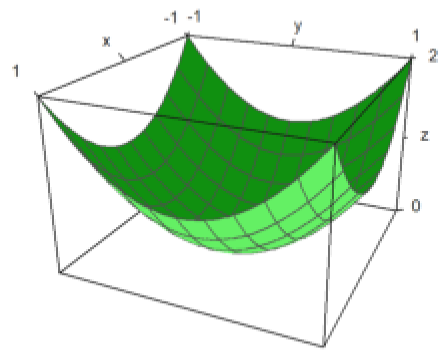
$$f(x,y) = x^2 + y^2$$

Maka perintah yang dapat dituliskan yaitu:

>function f(x,y)&= $x^{2+y}2$;

2) Panggil fungsinya

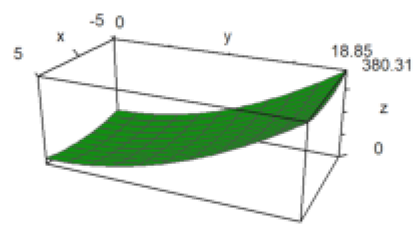
>plot3d("f(x,y)"):



Gambar 215: images/PLOT3D_Diva%20Nagita_23030630024-107.png

3) Tentukan rentang variabelnya

>plot3d("f(x,y)",-5,5,0,6*pi):



Gambar 216: images/PLOT3D_Diva%20Nagita_23030630024-108.png

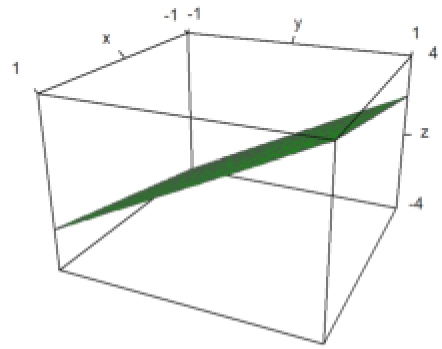
Membuat Grafik Fungsi Linear

$$g(x, y) = x + 3y$$

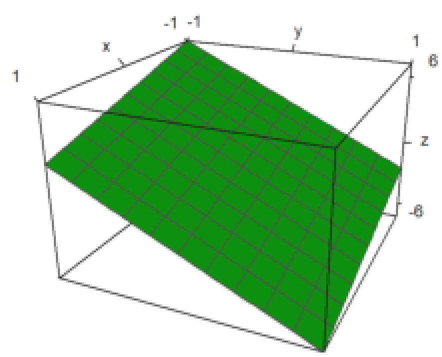
>function g(x,y) &= x+3*y;
>plot3d("g(x,y)"):

$$M(x, y) = -2x - 4y$$

>function M(x,y) &= -2*x-4*y;
>plot3d("M(x,y)"):

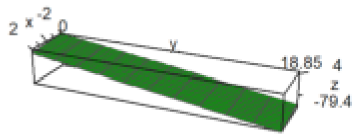


Gambar 217: images/PLOT3D_Diva%20Nagita_23030630024-110.png



Gambar 218: images/PLOT3D_Diva%20Nagita_23030630024-112.png

Rentang variabel:
>plot3d("M(x,y)",-2,2,0,6*pi):

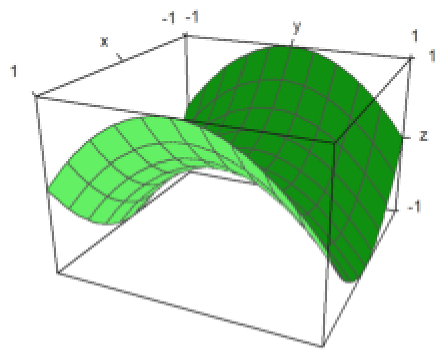


Gambar 219: images/PLOT3D_Diva%20Nagita_23030630024-113.png

Membuat Grafik Fungsi Kuadrat

$$Q(x, y) = x^2 - y^2$$

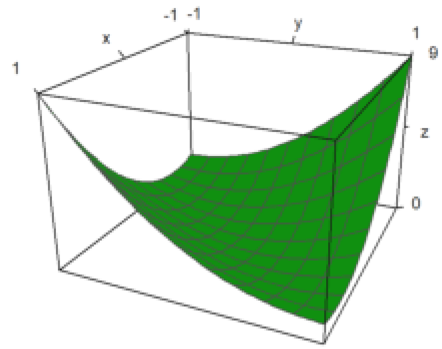
>function Q(x,y) &= x^2 - y^2;
>plot3d("Q(x,y)"):



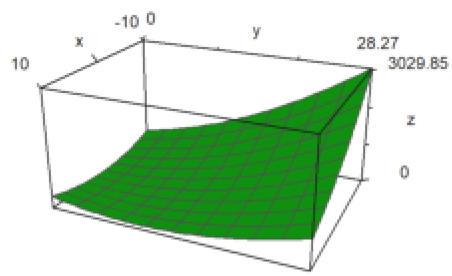
Gambar 220: images/PLOT3D_Diva%20Nagita_23030630024-115.png

$$P(x, y) = 3x^2 - 4xy + 2y^2$$

>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):



Gambar 221: images/PLOT3D_Diva%20Nagita_23030630024-117.png

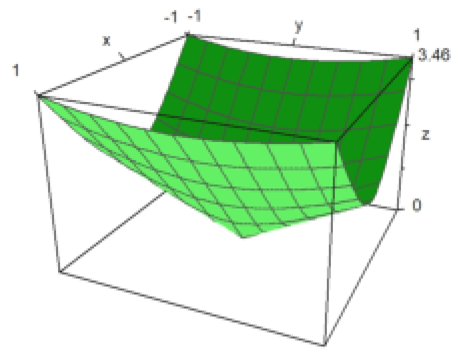


Gambar 222: images/PLOT3D_Diva%20Nagita_23030630024-118.png

Membuat Grafik Fungsi Akar Kuadrat

$$A(x, y) = \sqrt{10x^2 + 2y^2}$$

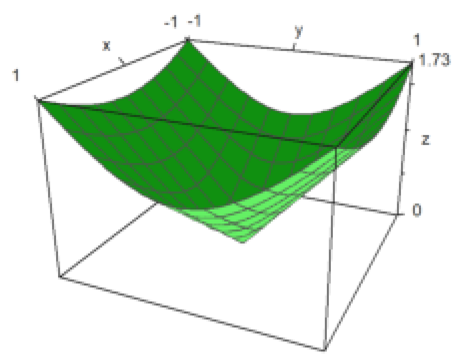
>function A(x,y) &= sqrt(10*x^{2+2*y}2);
>plot3d("A"):



Gambar 223: images/PLOT3D_Diva%20Nagita_23030630024-120.png

$$W(x, y) = \sqrt{x^2 + 2y^2}$$

>function W(x,y) &= sqrt(x^{2+2*y}2);
>plot3d("W"):



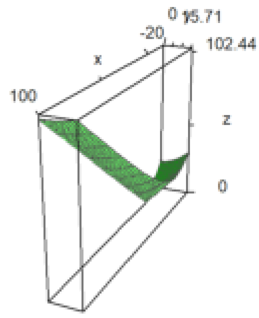
Gambar 224: images/PLOT3D_Diva%20Nagita_23030630024-122.png

Rentang Variabel:
>plot3d("W(x,y)",-20,100,0,5*pi):

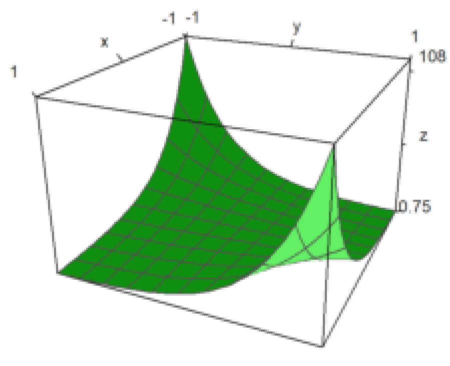
Membuat Grafik Fungsi Eksponensial

$$F(x, y) = 9.12^{xy}$$

>function F(x,y) &= 9*12^(x*y);



Gambar 225: images/PLOT3D_Diva%20Nagita_23030630024-123.png

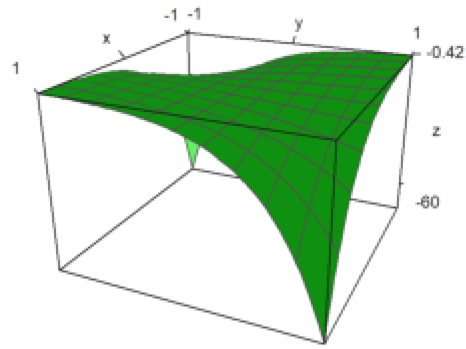


Gambar 226: images/PLOT3D_Diva%20Nagita_23030630024-125.png

>plot3d("F"):

$$H(x,y) = 5.(-20)^{xy}$$

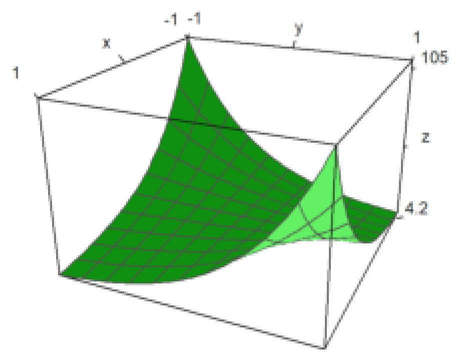
>function H(x,y) &= 5*-12^(x*y);
>plot3d("H"):



Gambar 227: images/PLOT3D_Diva%20Nagita_23030630024-127.png

$$T(x,y) = (-21).5^{xy}$$

>function T(x,y) &= -21*-5^(x*y);
>plot3d("T"):

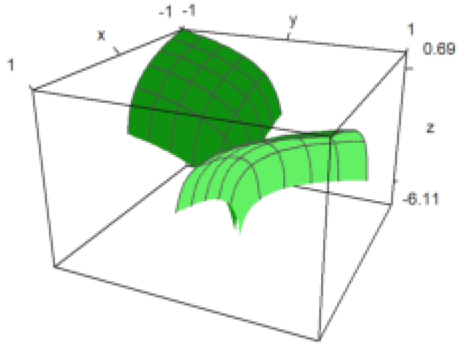


Gambar 228: images/PLOT3D_Diva%20Nagita_23030630024-129.png

Membuat Grafik Fungsi Logaritma

$$B(x,y) = log(x.2y), Basis10$$

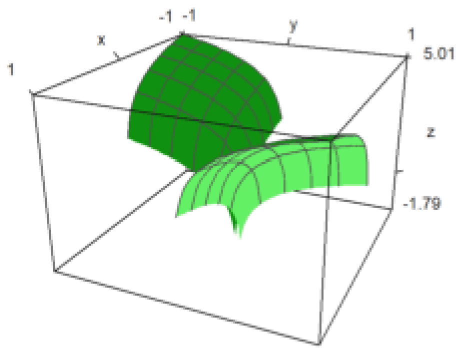
>function B(x,y) &= log(x*2*y);
>plot3d("B"):



Gambar 229: images/PLOT3D_Diva%20Nagita_23030630024-131.png

$$C(x, y) = log(30x.5y), basis10$$

>function C(x,y) &= log(30*x*5*y);
>plot3d("C"):



Gambar 230: images/PLOT3D_Diva%20Nagita_23030630024-133.png

Membuat Grafik Fungsi Trigonometri

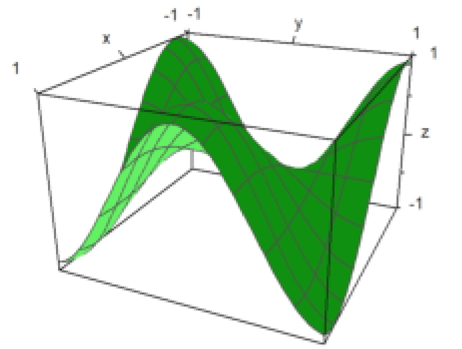
$$D(x, y) = sin(2x).cos(3y)$$

>function D(x,y) &= sin(2*x)*cos(3*y);
>plot3d("D"):

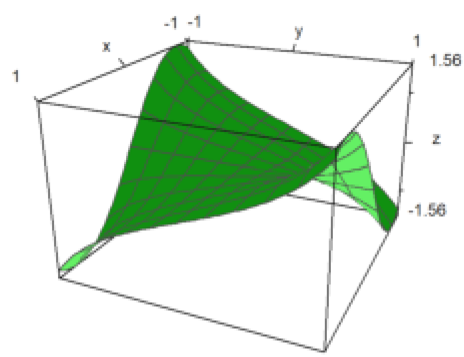
$$J(x, y) = sin(2x).tan(y)$$

>function J(x,y) &= sin(2*x)*tan(y);
>plot3d("J"):

$$G(x, y) = sec(2x).cot(5y)$$

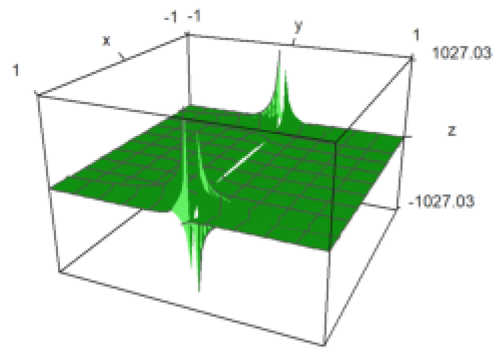


Gambar 231: images/PLOT3D_Diva%20Nagita_23030630024-135.png



Gambar 232: images/PLOT3D_Diva%20Nagita_23030630024-137.png

>function G(x,y) &= sec(2*x)*cot(y);
>plot3d("G"):

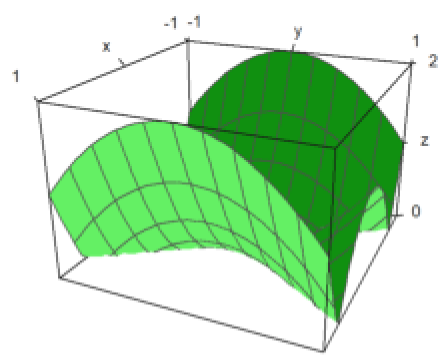


Gambar 233: images/PLOT3D_Diva%20Nagita_23030630024-139.png

Membuat Grafik Fungsi Nilai Mutlak

$$T(x,y) = |2x^2 - y^2|$$

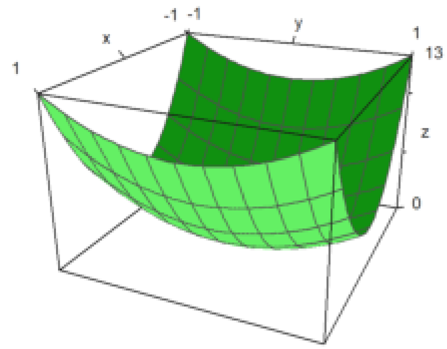
>function T(x,y) &= abs(2*x^2 - y^2);
>plot3d("T"):



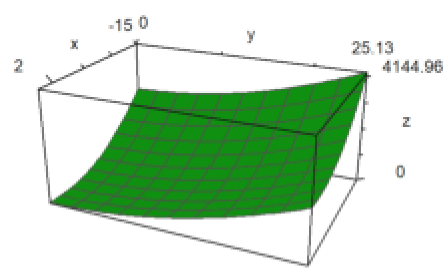
Gambar 234: images/PLOT3D_Diva%20Nagita_23030630024-141.png

$$M(x,y) = |-10x^2 - 3y^2|$$

>function M(x,y) &= abs(-10*x^2 - 3*y^2);
>plot3d("M"):
Rentang Variabel :
>plot3d("M(x,y)",-15,2,0,8*pi):



Gambar 235: images/PLOT3D_Diva%20Nagita_23030630024-143.png



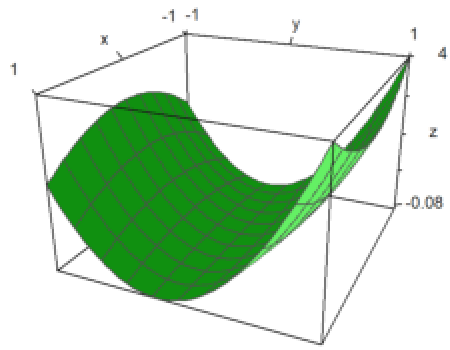
Gambar 236: images/PLOT3D_Diva%20Nagita_23030630024-144.png

Latihan

Buatlah grafik dari fungsi berikut:

$$A(x, y) = x^2y + 3y^2$$

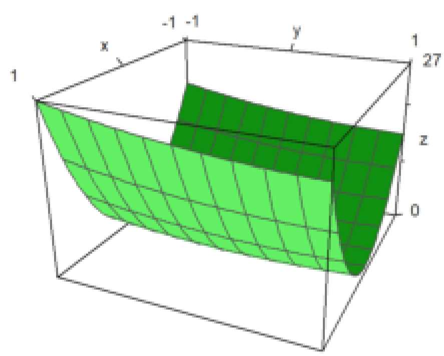
>function A(x,y) &= x$^{2*y+3*y}$2;
>plot3d ("A"):



Gambar 237: images/PLOT3D_Diva%20Nagita_23030630024-146.png

$$B(x, y) = y^2 - 2x^2y + 4x^3 + 20x^2$$

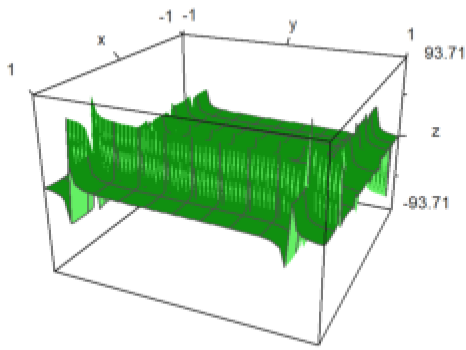
>function B(x,y)&= y^{2-2*x}2*y+4*x^{3+20*x}2;
>plot3d("B"):



Gambar 238: images/PLOT3D_Diva%20Nagita_23030630024-148.png

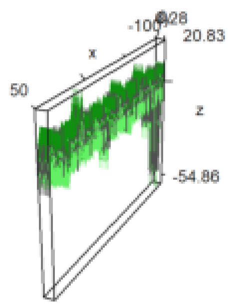
$$C(x, y) = cosec(9x) - tan(2y)$$

>function C(x,y)&= cosec(9*x)-tan(2*y);
>plot3d ("C"):
Beri rentang variabel untuk fungsi C(x,y):



Gambar 239: images/PLOT3D_Diva%20Nagita_23030630024-150.png

-100,50,0,2*pi
>plot3d("C(x,y)",-100,50,0,2*pi):



Gambar 240: images/PLOT3D_Diva%20Nagita_23030630024-151.png

Menggambar Data x, y, z pada ruang Tiga Dimensi (3D)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).

$$\gamma(t, s) = (x(t, s), y(t, s), z(t, s))$$

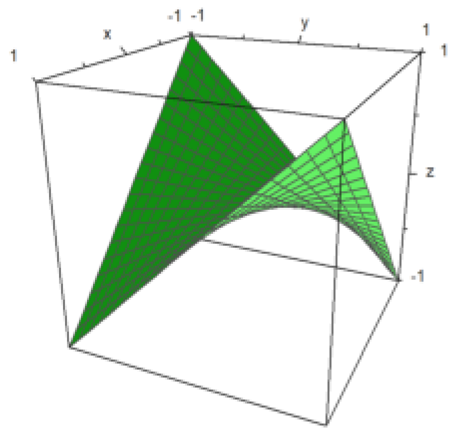
Karena x,y,z adalah matriks, kita asumsikan bahwa (t,s) melalui sebuah kotak persegi. Hasilnya, kita dapat memplot gambar persegi panjang di ruang angkasa.

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

>t=-1:0.1:1; s=(-1:0.1:1)'; plot3d(t,s,t*s):



Gambar 241: images/PLOT3D_Diva%20Nagita_23030630024-153.png

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

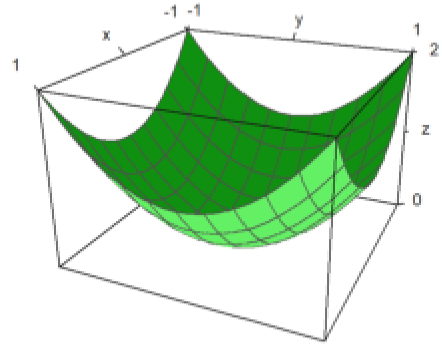
>plot3d("x^{2+y}2",grid= 2,fillcolor=[blue,blue]):

Contoh 3

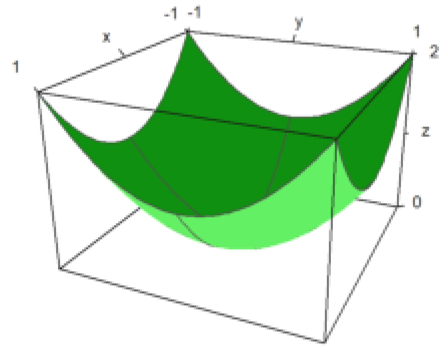
Jika kita ingin membuat plot 3d pada fungsi

$$2x^2 + y^3$$

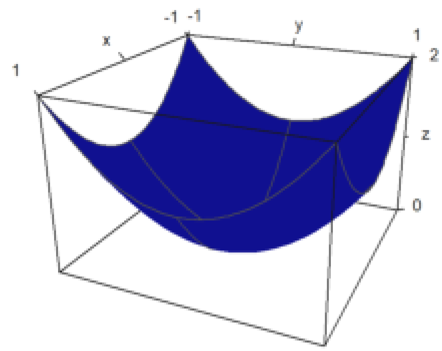
kita bisa menggunakan perintah seperti dibawah ini



Gambar 242: images/PLOT3D_Diva%20Nagita_23030630024-155.png

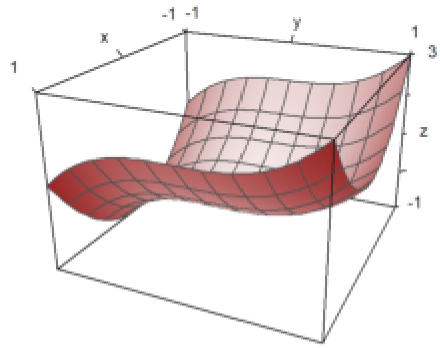


Gambar 243: images/PLOT3D_Diva%20Nagita_23030630024-156.png



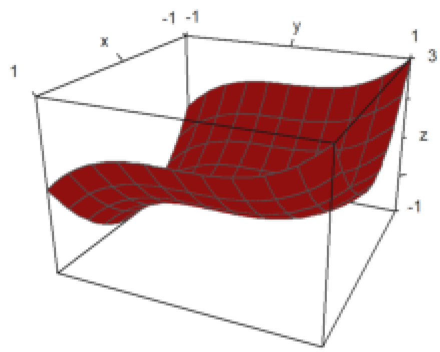
Gambar 244: images/PLOT3D_Diva%20Nagita_23030630024-157.png

>plot3d(“2x^{2+y}3”,grid=10,>hue, color=red);
>insimg()



Gambar 245: images/PLOT3D_Diva%20Nagita_23030630024-159.png

Jika kita mau menebalkan warna pada gambar diatas makam bisa menggunakan perintah
>plot3d(“2x^{2+y}3”,grid=10,fillcolor=[red,red]);
>insimg()



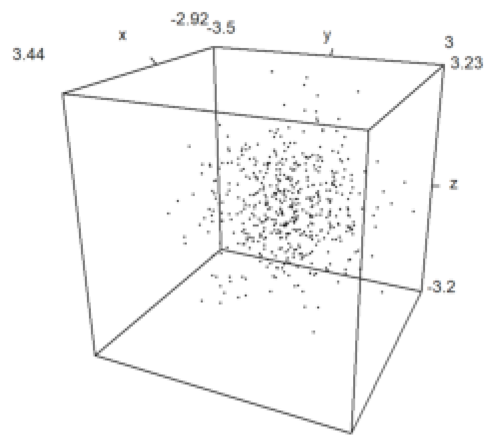
Gambar 246: images/PLOT3D_Diva%20Nagita_23030630024-160.png

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 5

Dalam contoh berikut, kita membuat tampilan bayangan dari bola yang



Gambar 247: images/PLOT3D_Diva%20Nagita_23030630024-161.png

terdistorsi. Koordinat biasa untuk bola adalah

$$\gamma(t,s) = (\cos(t)\cos(s), \sin(t)\sin(s), \cos(s))$$

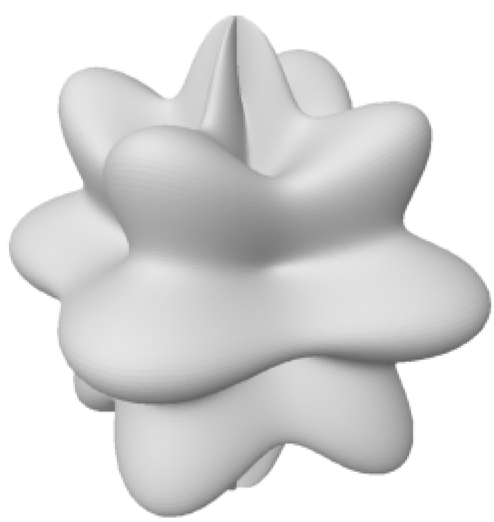
dengan

$$0 \le t \le 2\pi, \quad \frac{-\pi}{2} \le s \le \frac{\pi}{2}.$$

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):
```



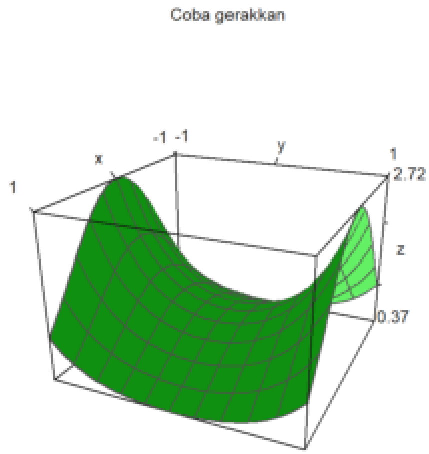
Gambar 248: images/PLOT3D_Diva%20Nagita_23030630024-165.png

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.

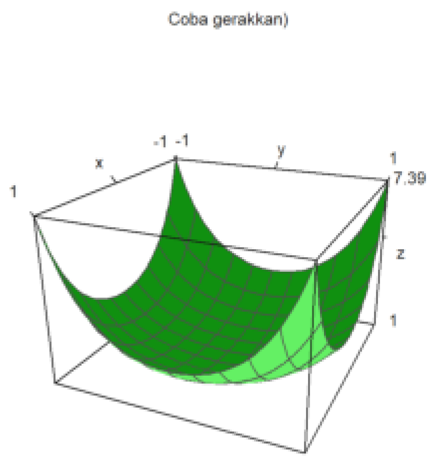
Interaksi user dimungkinkan dengan parameter >user. dengan perintah >user kita dapat menekan tombol berikut. + kiri, kanan, atas, bawah: memutar sudut pandang + +,-: memperbesar atau memperkecil + a: menghasilkan anaglyph (lihat di bawah) + l : tombol nyalakan sumber cahaya (lihat dibawah) + spasi: reset ke default + kembali: akhiri interaksi

>plot3d("exp(-x^{2+y}2)",>user,...
> title="Coba gerakkan"):



Gambar 249: images/PLOT3D_Diva%20Nagita_23030630024-166.png

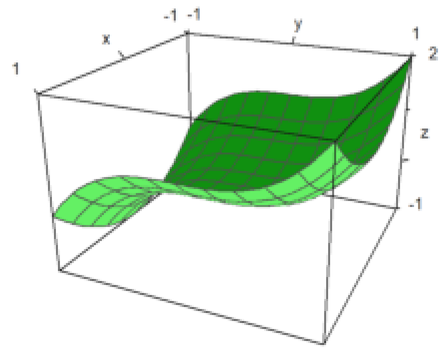
>plot3d("exp(x^{2+y}2)",>user,...
> title="Coba gerakkan)"):



Gambar 250: images/PLOT3D_Diva%20Nagita_23030630024-167.png

Animasi 3D

>function testplot () := plot3d(“x^{2+y}3”);. . .
> rotate(“testplot”); testplot():

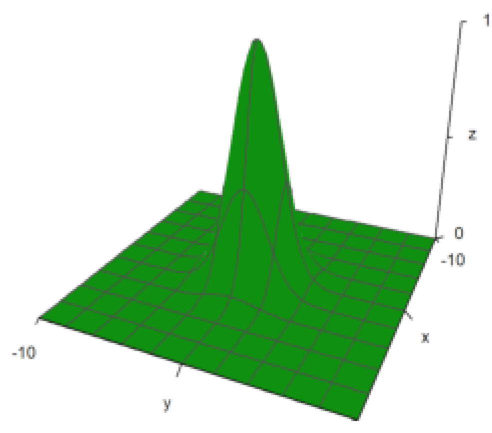


Gambar 251: images/PLOT3D_Diva%20Nagita_23030630024-168.png

Fungsi rotate yaitu untuk memutar plot.
Fungsi ini akan membuat animasi plot 3D dari fungsi

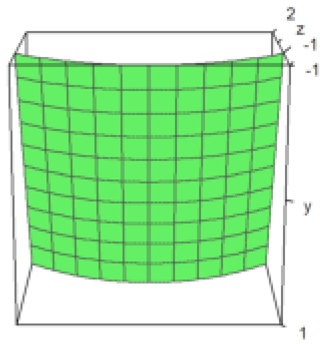
$$x^2 + y^3$$

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):



Gambar 252: images/PLOT3D_Diva%20Nagita_23030630024-170.png

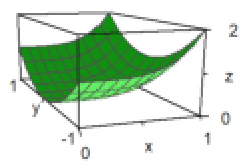
Ada beberapa parameter untuk menskalakan fungsi atau mengubah tampilan grafik.
fscale: menskalakan ke nilai fungsi (defaultnya adalah <fscale).
scale: angka atau vektor 1x2 untuk diskalakan ke arah x dan y.
frame: jenis bingkai (default 1).
>plot3d(“xˆ2+y”,distance=10,zoom=5,angle=0,height=5):



Gambar 253: images/PLOT3D_Diva%20Nagita_23030630024-171.png

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=2):



Gambar 254: images/PLOT3D_Diva%20Nagita_23030630024-172.png

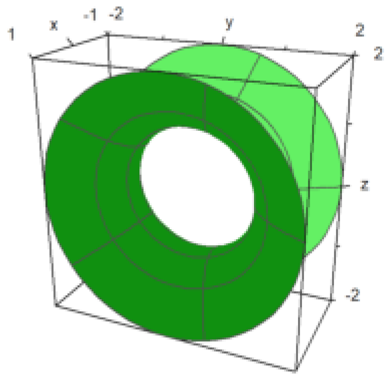
Plot selalu terlihat berada di tengah kubus plot. Kita dapat memindahkan bagian tengah dengan center parameter.

>plot3d(“xˆ2+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

Menggambar Fungsi Parametrik Tiga Dimensi (3D)

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.



Gambar 255: images/PLOT3D_Diva%20Nagita_23030630024-173.png

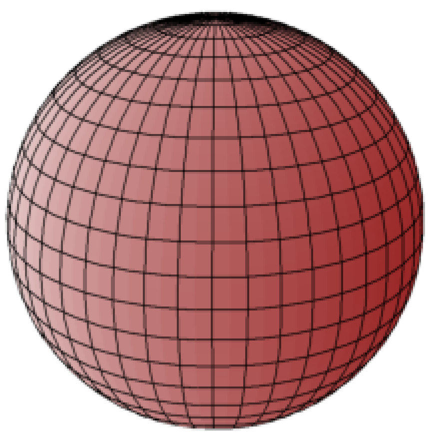
Fungsi parametrik merupakan salah satu cara mendefinisikan kurva atau permukaan dalam ruang 2D atau 3D menggunakan satu atau lebih parameter independen.

- 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),

$$x = f(t),$$
$$y = g(t),$$
$$z = h(t),$$

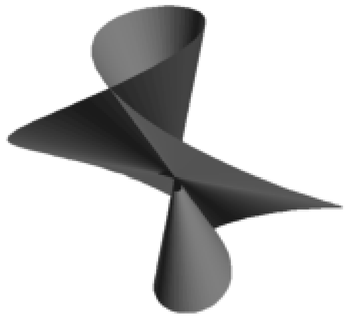
Contoh Soal

>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):



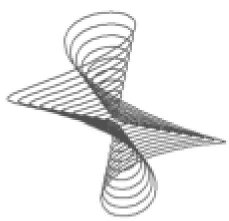
Gambar 256: images/PLOT3D_Diva%20Nagita_23030630024-177.png

>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):



Gambar 257: images/PLOT3D_Diva%20Nagita_23030630024-178.png

>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):



Gambar 258: images/PLOT3D_Diva%20Nagita_23030630024-179.png

>reset;. . .
> S:=normal(10,1250); plot3d(S[3],S[6],S[9],>points,style=".."):

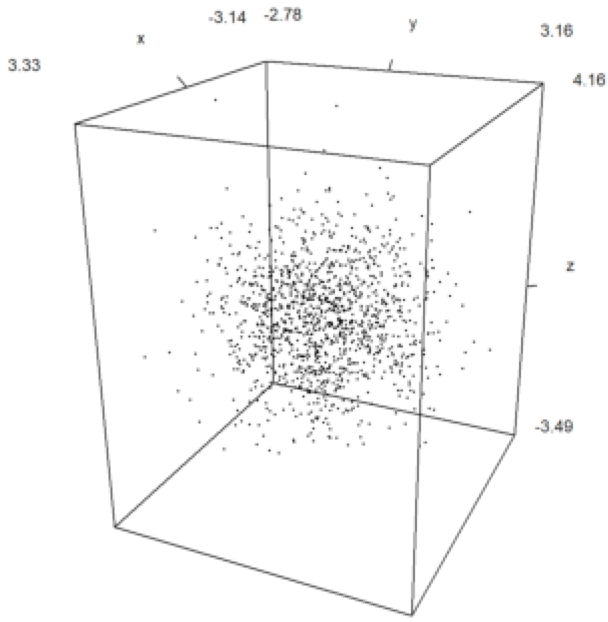
>S:=normal(10,1250); T:=cumsum(normal(10,1250));. . .
> plot3d(T[2],T[5],T[8],>wire,. . .
> linewidth=2,>anaglyph,zoom=3):

>P=cumsum(normal(5,75));. . .
> plot3d(P[3],P[4],P[5],>anaglyph,>wire):

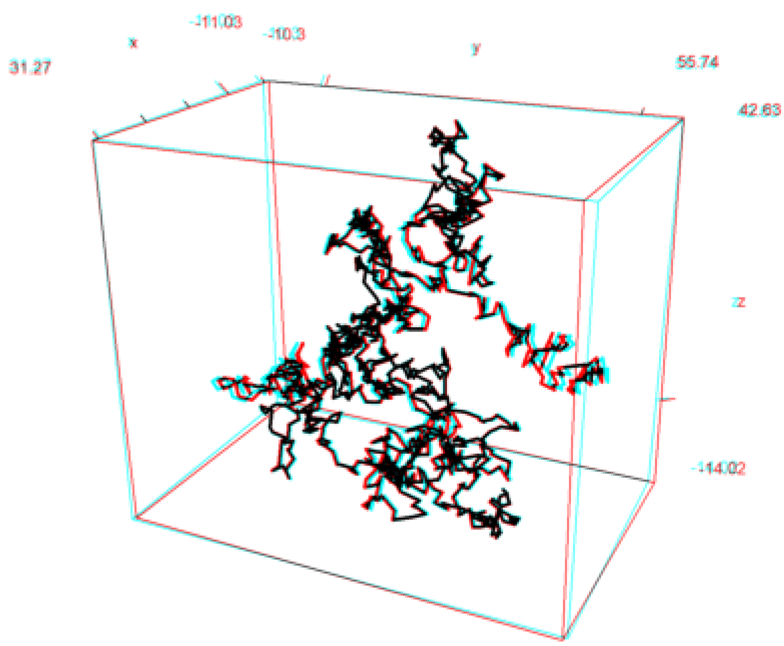
Menggambar Fungsi Implisit Tiga Dimensi (3D)

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.

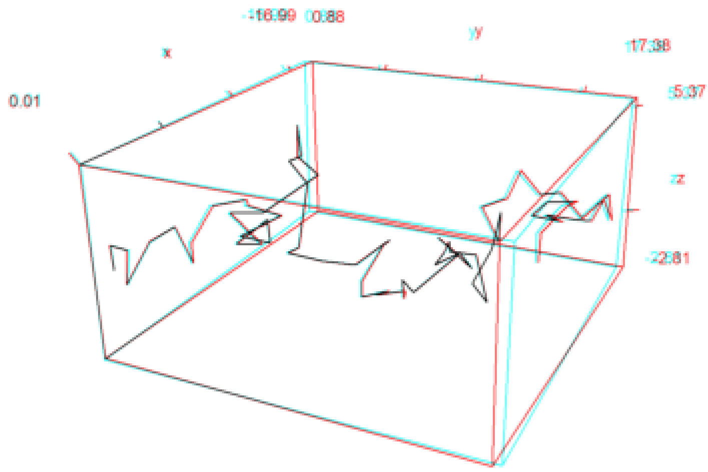
$$F(x,y,z)=0$$



Gambar 259: images/PLOT3D_Diva%20Nagita_23030630024-180.png



Gambar 260: images/PLOT3D_Diva%20Nagita_23030630024-181.png



Gambar 261: images/PLOT3D_Diva%20Nagita_23030630024-182.png

(1 persamaan dan 3 variabel), terdiri dari 2 variabel bebas dan 1 terikat

Misalnya, F(x, y, z) = x^2 + y^2 + z^2 - 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

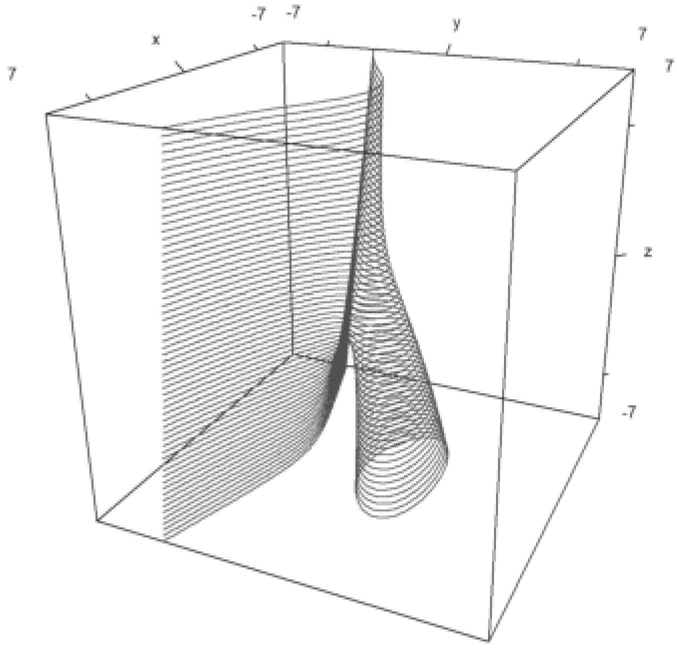
$$M=\{(x,y,z):x^2+y^3+zy=1\}$$

—

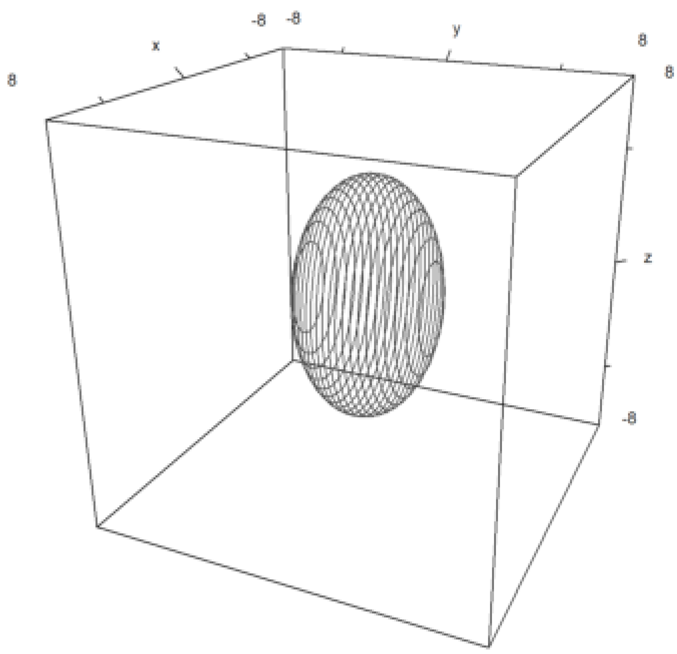
Contoh Fungsi Implisit

>plot3d(“x^{2+y}3+z*y-1”,r=7,implicit=4):

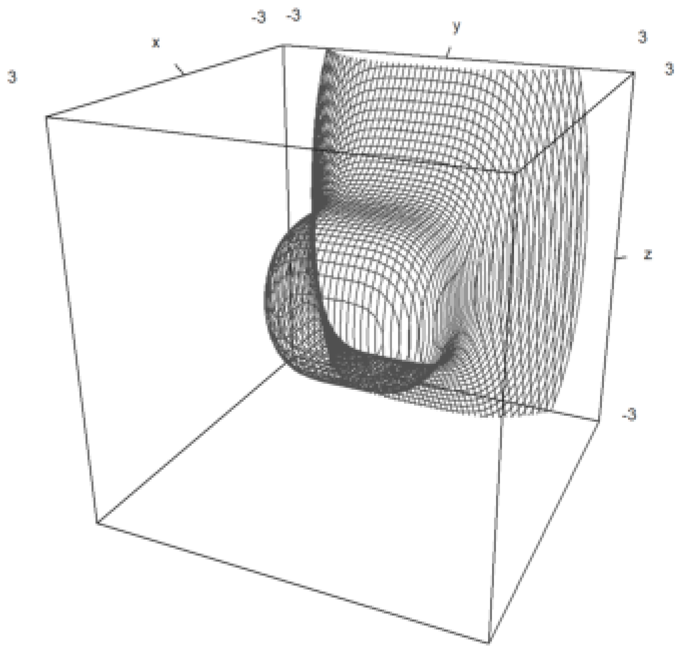
>plot3d(“2*x^2 + 3*y^2 + z^2 - 25”,r=8,implicit=2):



Gambar 262: images/PLOT3D_Diva%20Nagita_23030630024-186.png



Gambar 263: images/PLOT3D_Diva%20Nagita_23030630024-187.png



Gambar 264: images/PLOT3D_Diva%20Nagita_23030630024-188.png

>plot3d(“4*xˆ3 + 3*yˆ4 + 6*zˆ2 - 10”,r=3,implicit=3):
>plot3d(“xˆ5 + 5*yˆ3 + 3*zˆ2 - 5*x - 7*y - 5*z + 10”,r=5,implicit=2):
>plot3d(“xˆ2 + yˆ2 - zˆ2”,r=5,implicit=3):
>plot3d(“xˆ3 + 2*yˆ2 +3*zˆ3-4”,r=5,implicit=3):
>plot3d(“x^{2+y}2+zˆ2+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)}$2)-d”,r=2,<frame,>implicit,>user):
>c=1; d=1;
>plot3d(“((x^{2+y}2+c$^{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,<frame,>implicit,>user):
>plot3d(“x^{3+y}5+5*x*z+zˆ3”,>implicit,r=3,zoom=2):
>plot3d(“x^{2+y}2+4*x*z+zˆ3-2”,>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-zˆ2+2*x*y*z-0”,>implicit,r=5,zoom=2.5):

Latihan soal

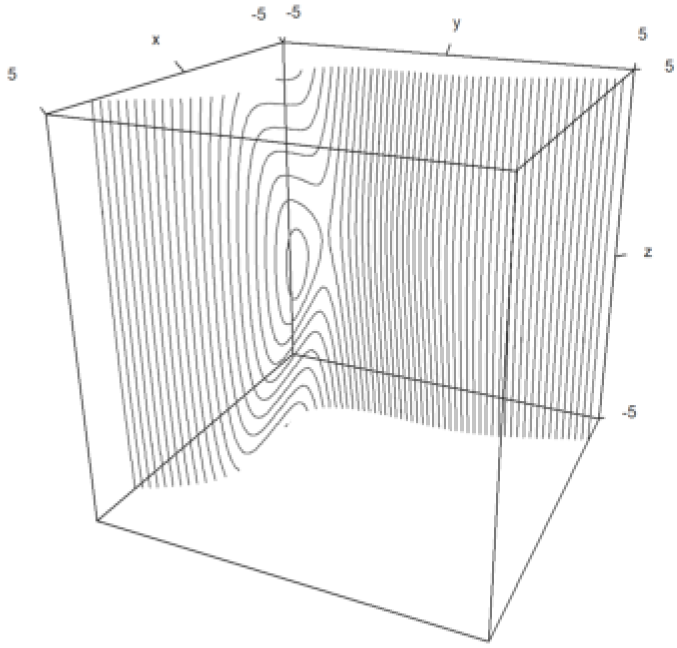
Gambarlah Fungsi implisit berikut dalam 3D

$$f(x,y,z) = x^2 + y^2 - z^2 - 1$$

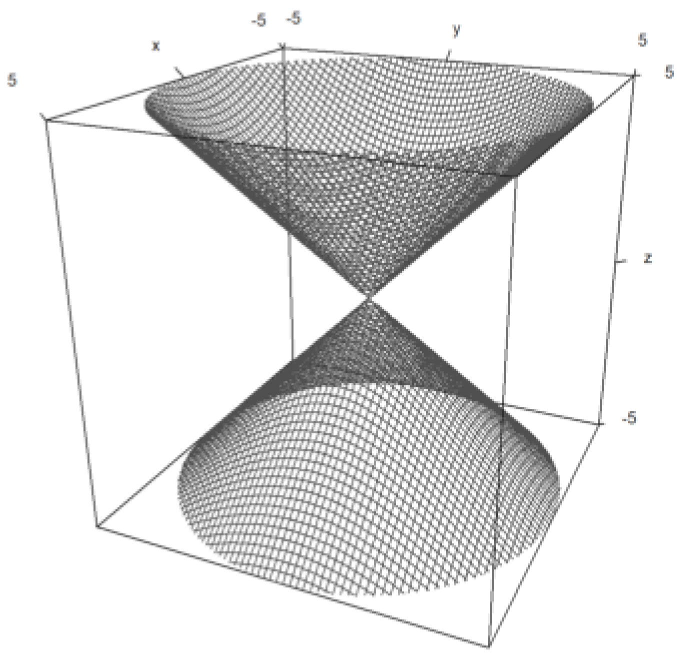
>plot3d(“x^{2+y}2-zˆ2-1”,r=8,implicit=3):
Gambarlah fungsi 3D dari fungsi implisit berikut ini

$$f(x,y,z) = xy + x^3y^2 + xz^3 - 9 = 0$$

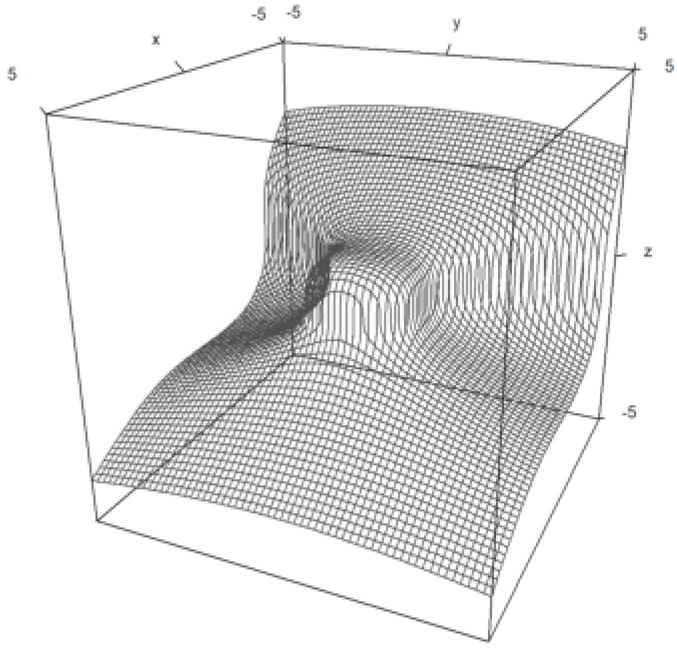
dengan r=4



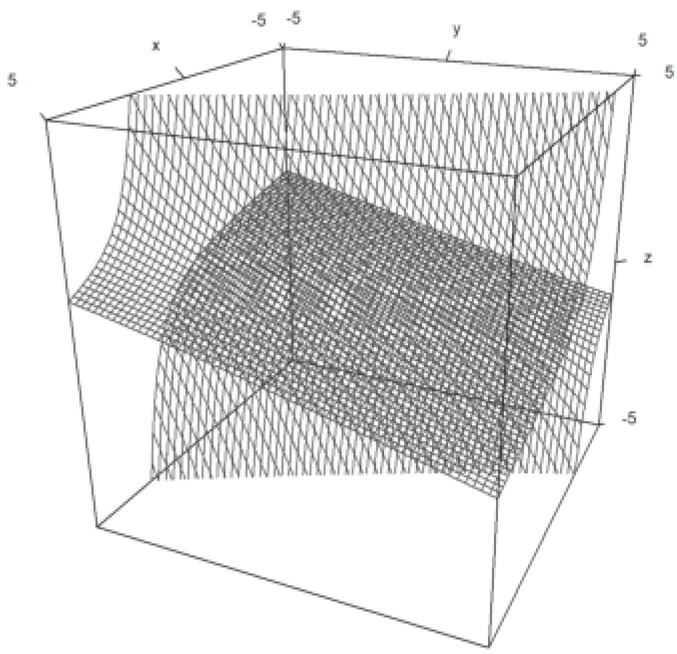
Gambar 265: images/PLOT3D_Diva%20Nagita_23030630024-189.png



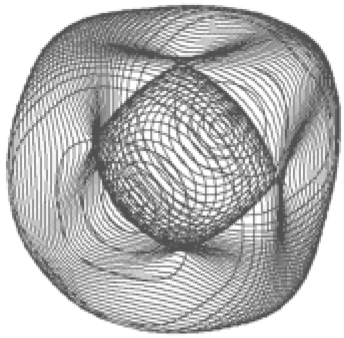
Gambar 266: images/PLOT3D_Diva%20Nagita_23030630024-190.png



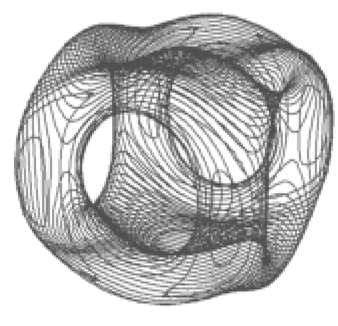
Gambar 267: images/PLOT3D_Diva%20Nagita_23030630024-191.png



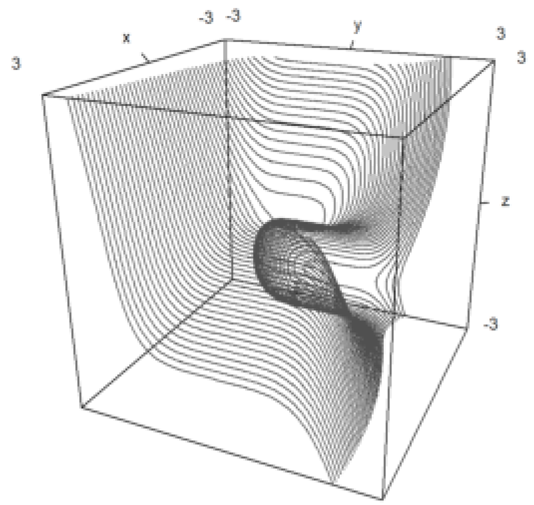
Gambar 268: images/PLOT3D_Diva%20Nagita_23030630024-192.png



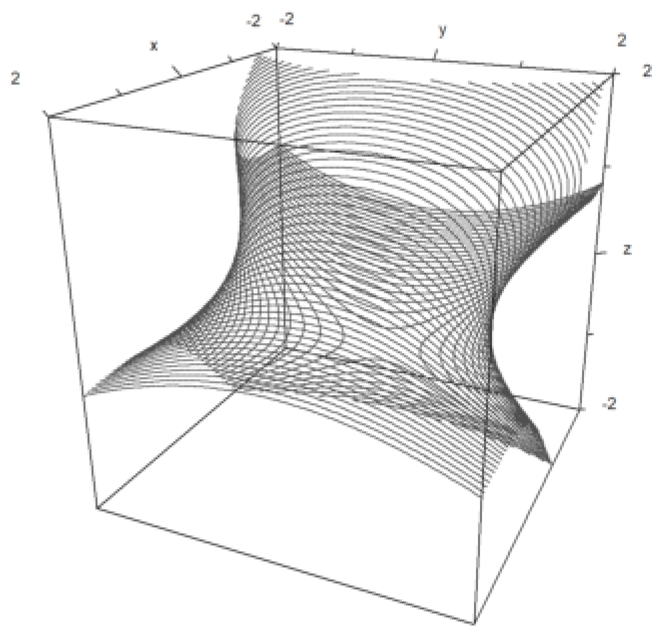
Gambar 269: images/PLOT3D_Diva%20Nagita_23030630024-193.png



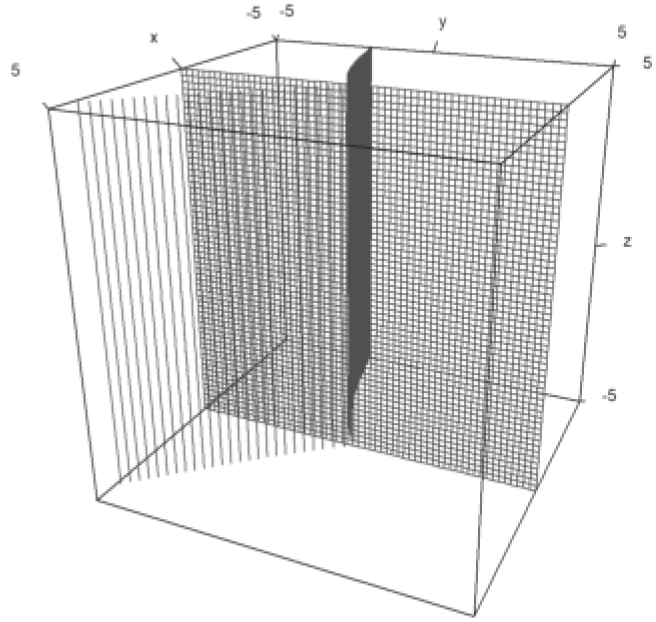
Gambar 270: images/PLOT3D_Diva%20Nagita_23030630024-194.png



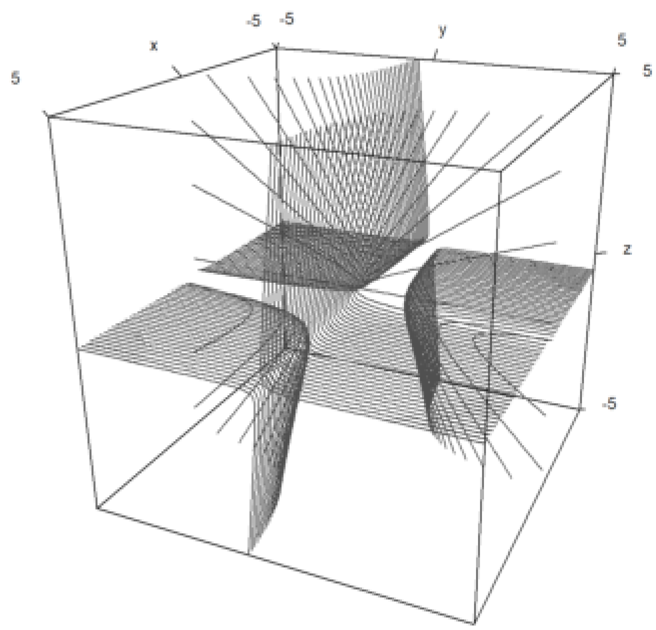
Gambar 271: images/PLOT3D_Diva%20Nagita_23030630024-195.png



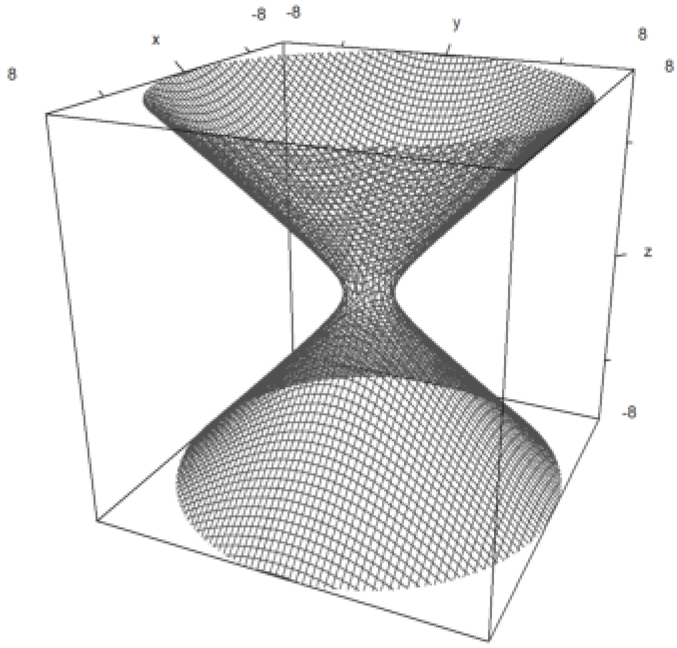
Gambar 272: images/PLOT3D_Diva%20Nagita_23030630024-196.png



Gambar 273: images/PLOT3D_Diva%20Nagita_23030630024-197.png



Gambar 274: images/PLOT3D_Diva%20Nagita_23030630024-198.png



Gambar 275: images/PLOT3D_Diva%20Nagita_23030630024-200.png

>plot3d(“x*y+x^{3*y}2+x*zˆ3-9”, r=4, implicit=3):

Mengatur Tampilan, Warna, dan Sudut Pandang Gambar Permukaan Tiga Dimensi (3D) dan Menampilkan Kontur serta Bidang kontur 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.

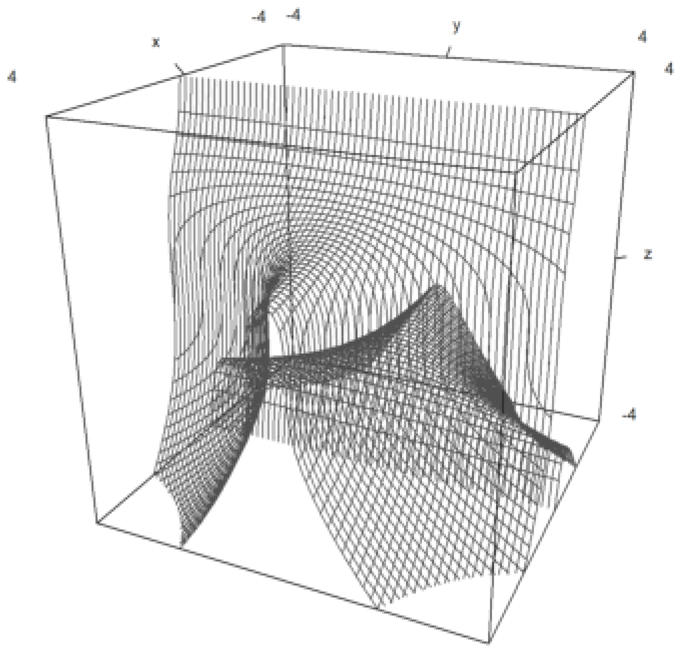
>plot3d(“exp(-x^{2-y}2)”,r=2,n=100,level=“thin”, ...
> >contour,>spectral,fscale=1,scale=1.1,angle=45°,height=20°):

>plot3d(“exp(x*y)”,angle=100°,>contour,color=green):

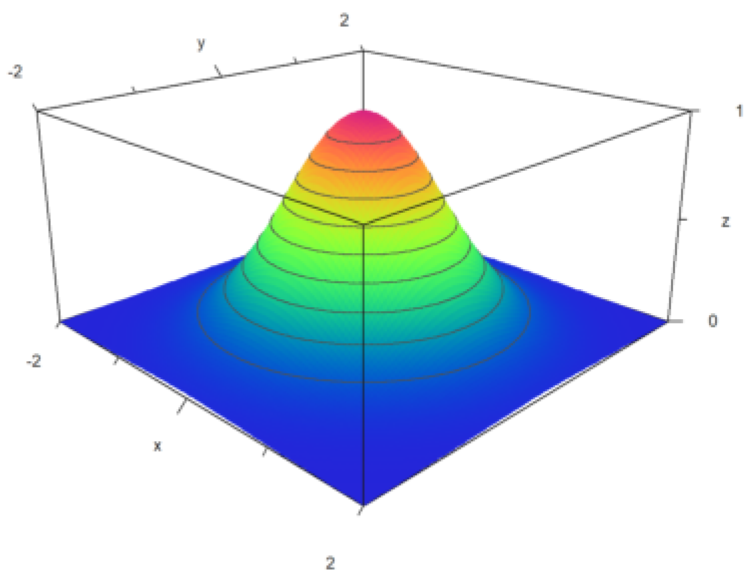
Bayangan default menggunakan warna abu-abu. Tetapi rentang warna spektral juga tersedia.

-> spektral: Menggunakan skema spektral default

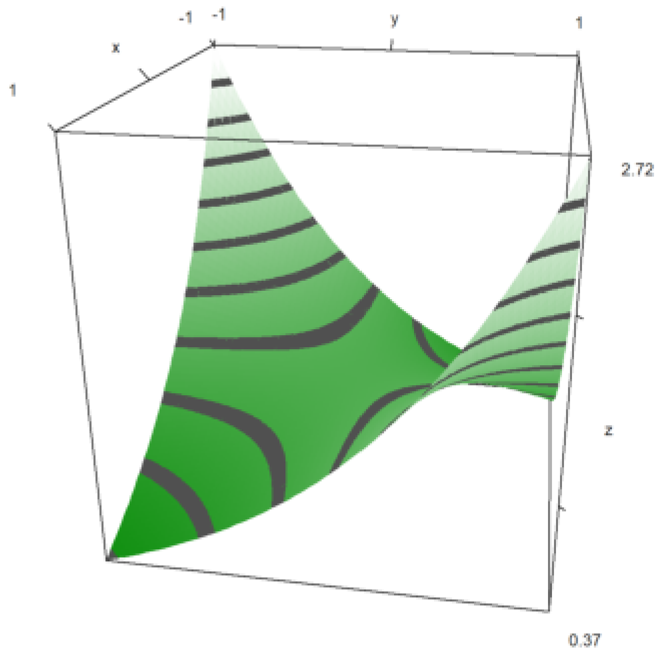
- color=...: Menggunakan warna khusus atau skema spektral



Gambar 276: images/PLOT3D_Diva%20Nagita_23030630024-202.png



Gambar 277: images/PLOT3D_Diva%20Nagita_23030630024-203.png



Gambar 278: images/PLOT3D_Diva%20Nagita_23030630024-204.png

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.

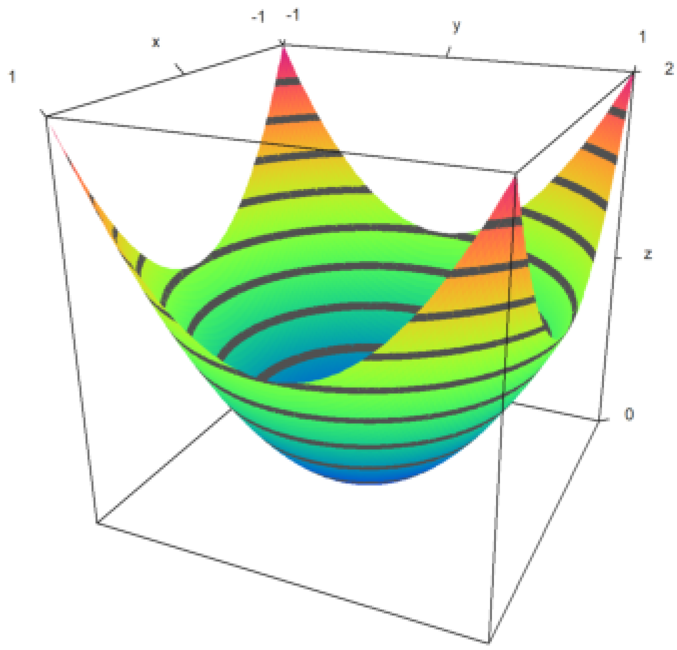
>plot3d("x^{y-y}x",level=0,a=0,b=6,c=0,d=6,contourcolor=red,n=100):

Dimungkinkan untuk menunjukkan bidang kontur di bawah plot. Warna dan jarak ke plot dapat ditentukan.

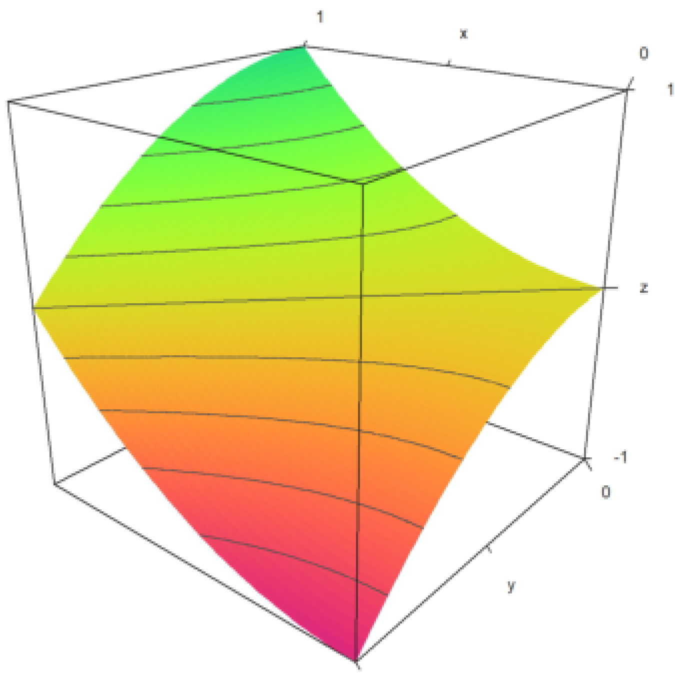
>plot3d("x^{2+y}4",>cp,cpcolor=green,cpdelta=0.2):

Berikut adalah beberapa gaya lagi. Kami selalu mematikan frame, dan menggunakan berbagai skema warna untuk plot dan grid.

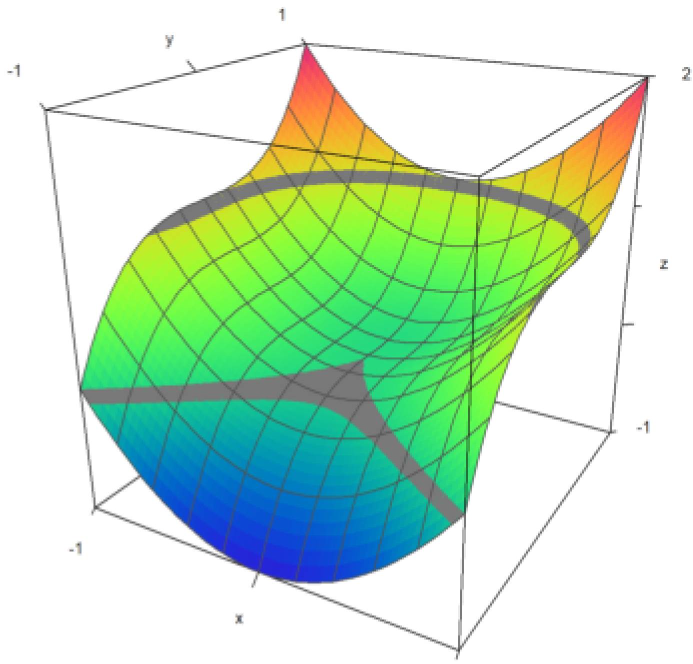
>figure(2,2); . . .
> expr="y^{3-x}2"; . . .
> figure(1); . . .
> plot3d(expr,<frame,>cp,cpcolor=spectral); . . .
> figure(2); . . .
> plot3d(expr,<frame,>spectral,grid=10,cp=2); . . .



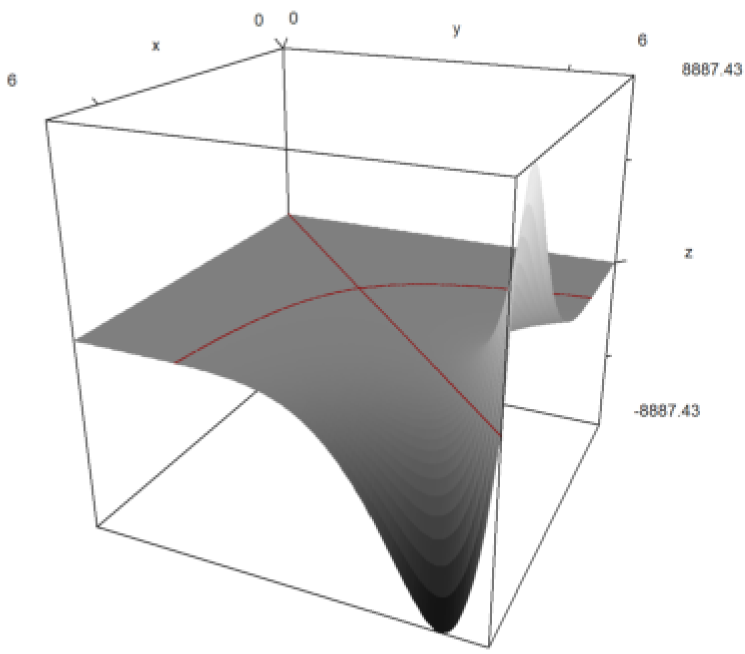
Gambar 279: images/PLOT3D_Diva%20Nagita_23030630024-205.png



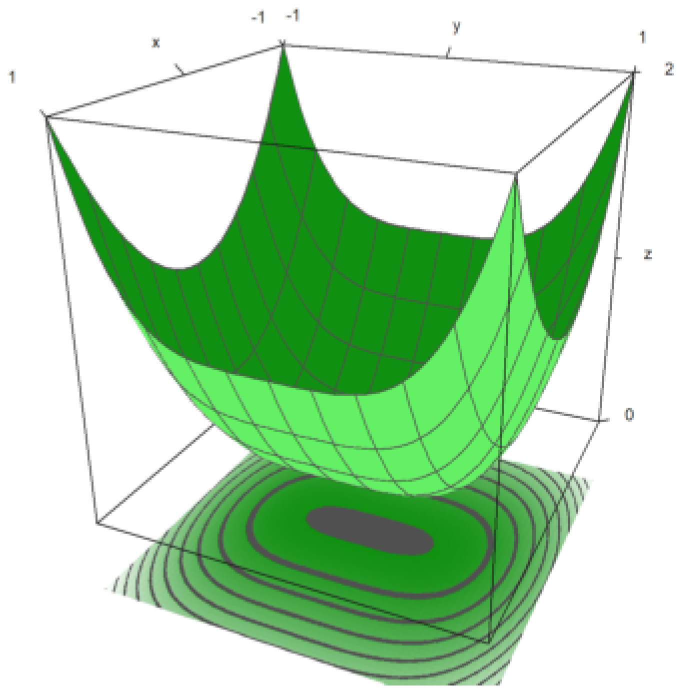
Gambar 280: images/PLOT3D_Diva%20Nagita_23030630024-206.png



Gambar 281: images/PLOT3D_Diva%20Nagita_23030630024-207.png



Gambar 282: images/PLOT3D_Diva%20Nagita_23030630024-209.png



Gambar 283: images/PLOT3D_Diva%20Nagita_23030630024-210.png

> 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

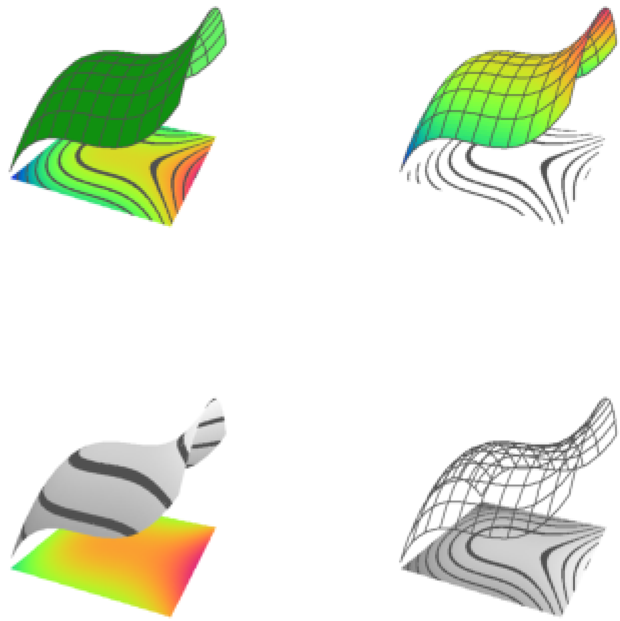
>figure(3,3); . . .
> for i=1:9; . . .
> figure(i); plot3d("$x^{2+y}2$",spectral=i,>contour,>cp,<frame,zoom=4); . . .
> end; . . .
> figure(0):

Sumber cahaya dapat diubah dengan l dan tombol kursor selama interaksi pengguna. Itu juga dapat diatur dengan parameter.

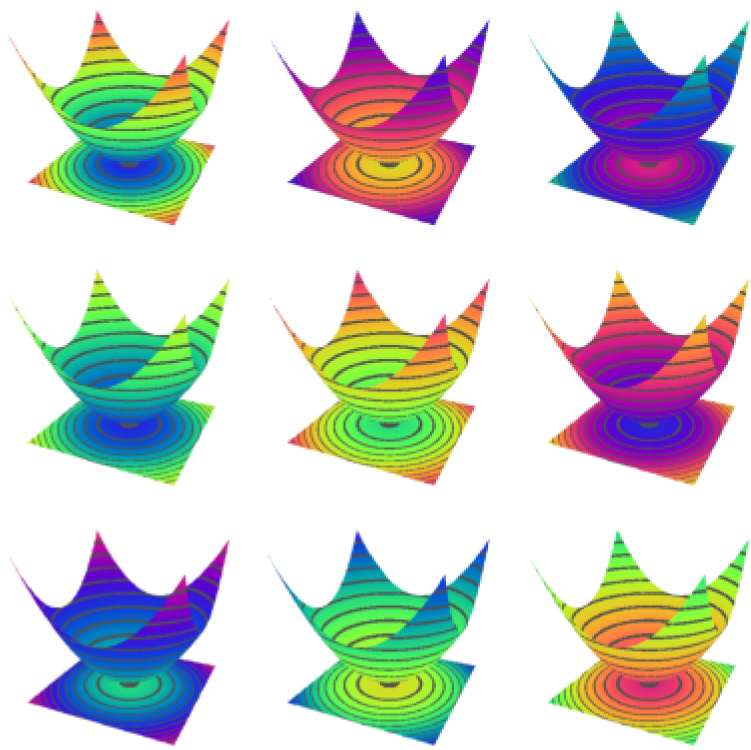
- cahaya: arah untuk cahaya
- amb: cahaya sekitar antara 0 dan 1

Perhatikan bahwa program tidak membuat perbedaan antara sisi plot. Tidak ada bayangan. Untuk ini, Anda perlu Povray.

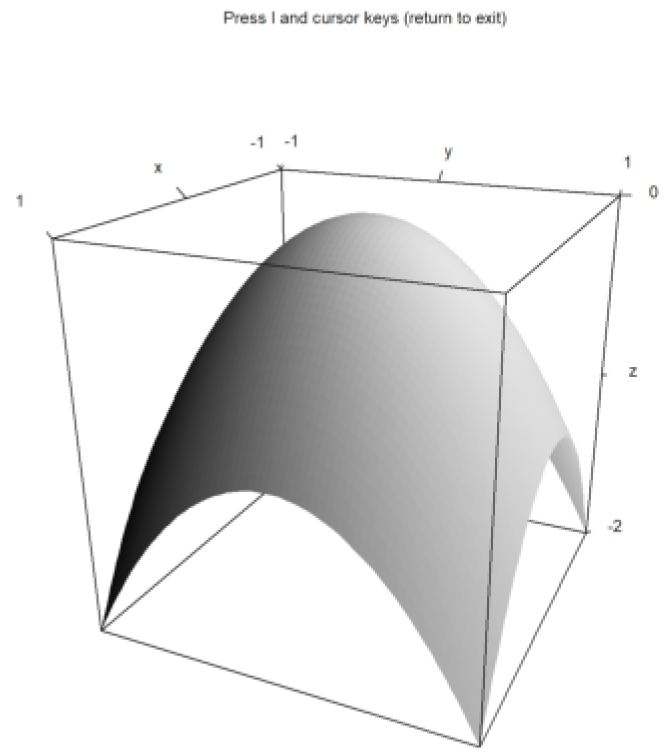
>plot3d("-$x^{2-y}2$", . . .
> hue=true,light=[0,1,1],amb=0,user=true, . . .
> title="Press l and cursor keys (return to exit)"):



Gambar 284: images/PLOT3D_Diva%20Nagita_23030630024-211.png



Gambar 285: images/PLOT3D_Diva%20Nagita_23030630024-212.png



Gambar 286: images/PLOT3D_Diva%20Nagita_23030630024-213.png

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):

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.

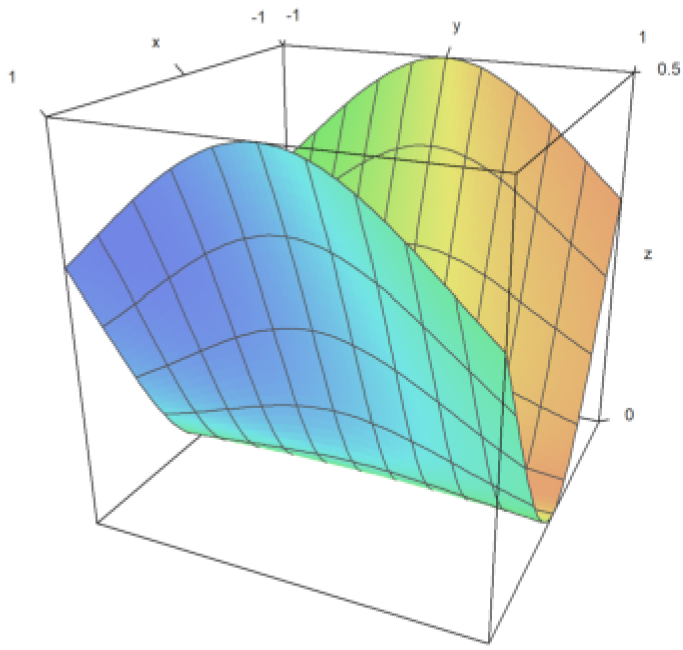
Dalam contoh ini, pertama-tama kita menghitung nilainya. Kemudian kita sesuaikan x dan y, sehingga vektor-vektornya berpusat pada nilai yang digunakan.

>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):

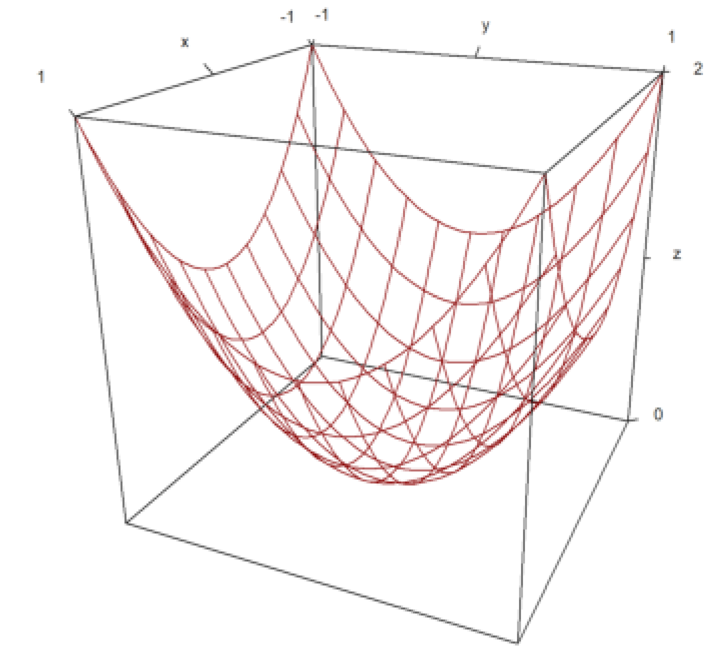
>x=-0.01:0.1:1; y=x'; z=x+2/3*y; ...
> xa=(x|1.1)-0.05; ya=(y_1.1)-0.05; ...
> plot3d(xa,ya,z,bar=true):

>x=-0.01:0.1:1; y=x'; z=1/2*x+1/2*y; ...

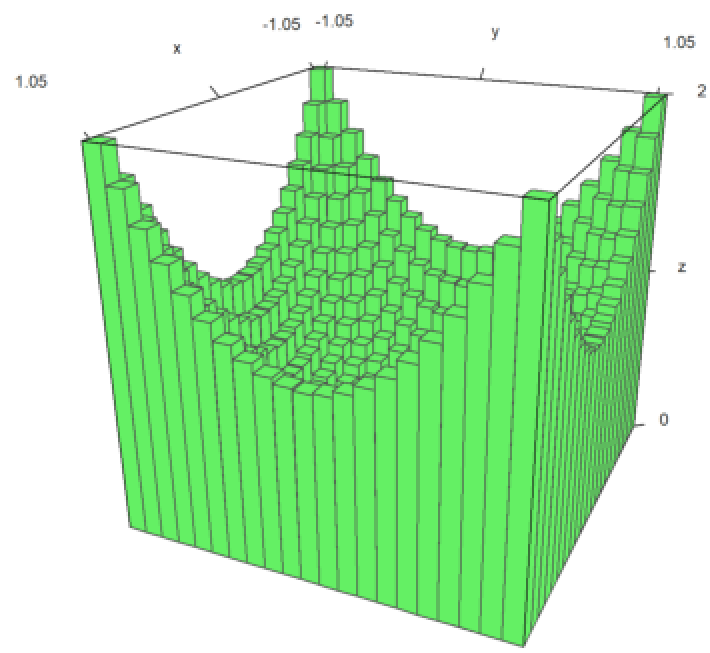
Gambar 287: images/PLOT3D_Diva%20Nagita_23030630024-214.png



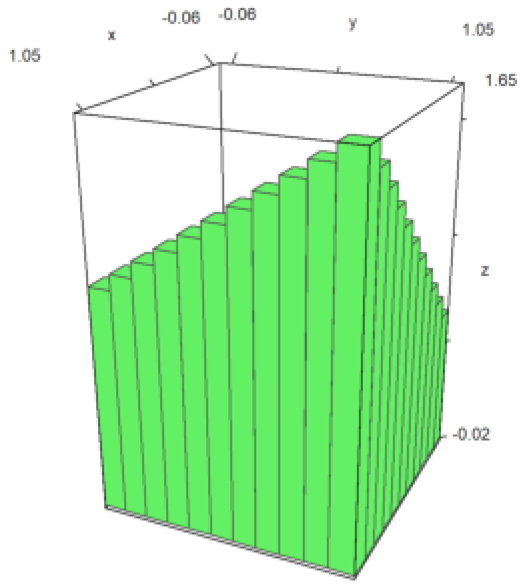
Gambar 288: images/PLOT3D_Diva%20Nagita_23030630024-215.png



Gambar 289: images/PLOT3D_Diva%20Nagita_23030630024-216.png



Gambar 290: images/PLOT3D_Diva%20Nagita_23030630024-217.png



Gambar 291: images/PLOT3D_Diva%20Nagita_23030630024-218.png

> 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):

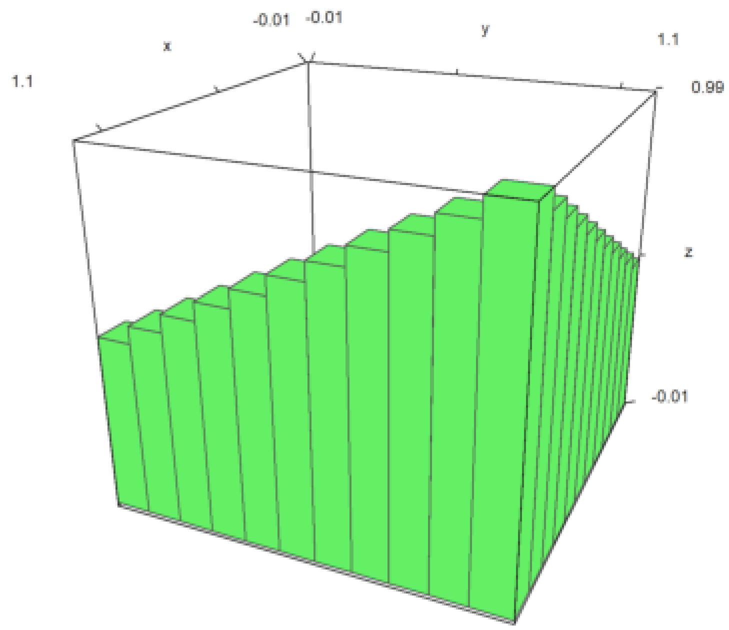
>Z=intrandom(5,100,6); v=zeros(5,6); ...
> loop 1 to 5; v[#]=getmultiplicities(1:6,Z[#]); end; ...
> columnsplot3d(v',scols=1:5,ccols=[1:5]):

>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]):

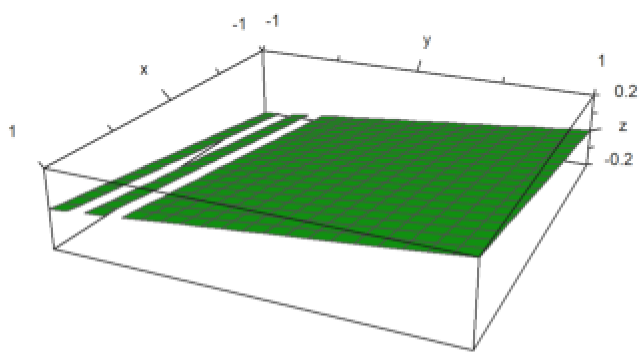
>Z=intrandom(7,1000,6); v=zeros(7,1); ...
> loop 1 to 7; v[#]=getmultiplicities(1:1,Z[#]); end; ...
> columnsplot3d(v',scols=1:7,ccols=[1:7]):

Menggambar Permukaan Benda Putar

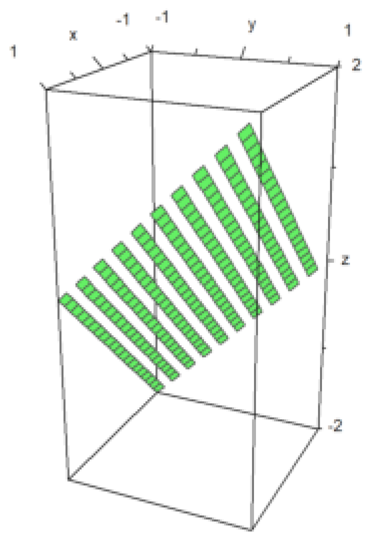
>plot2d("(x^{2+y}2-1)$^{3-x}$2*y^3",r=1.3, ...
> style="#",color=red,<outline, ...



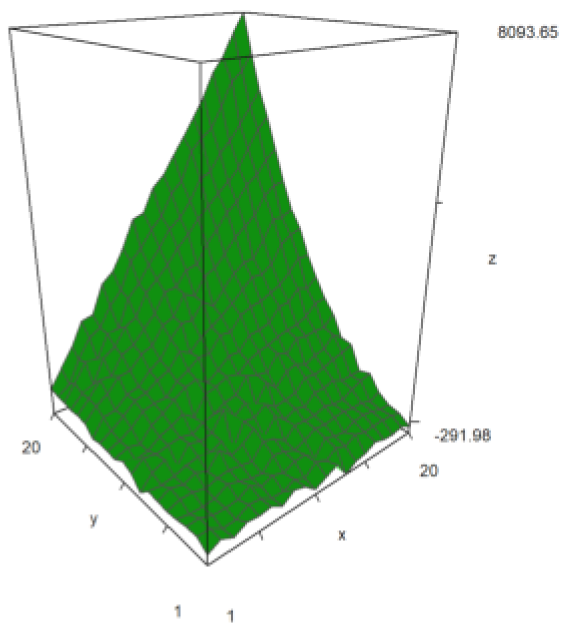
Gambar 292: images/PLOT3D_Diva%20Nagita_23030630024-219.png



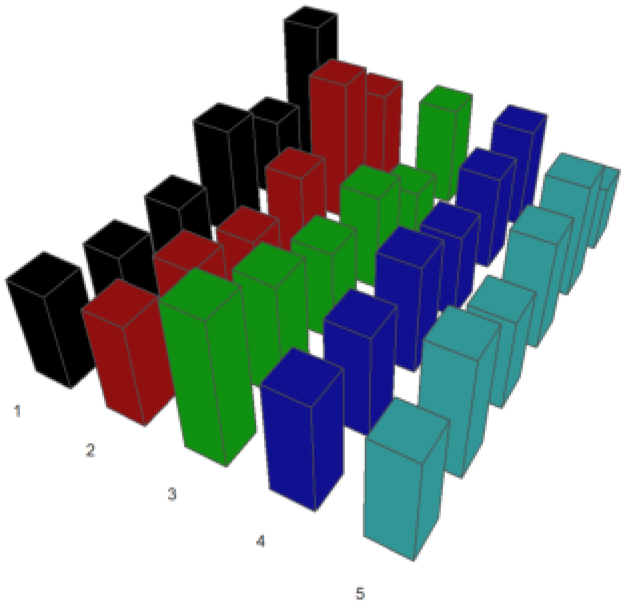
Gambar 293: images/PLOT3D_Diva%20Nagita_23030630024-220.png



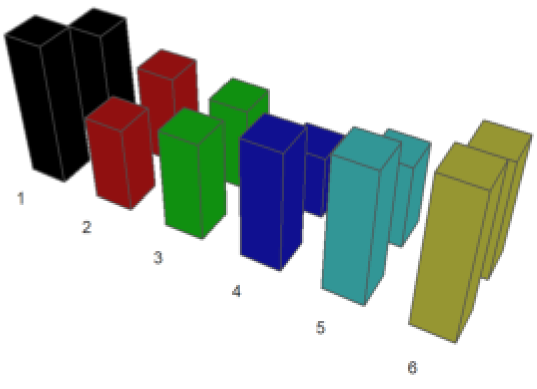
Gambar 294: images/PLOT3D_Diva%20Nagita_23030630024-221.png



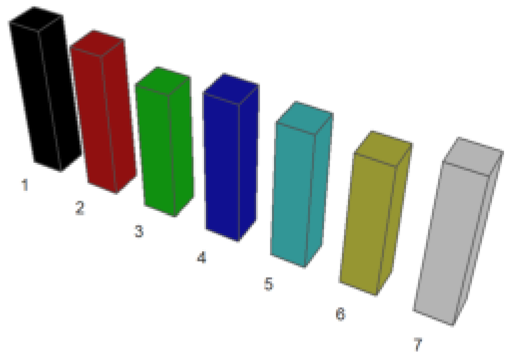
Gambar 295: images/PLOT3D_Diva%20Nagita_23030630024-222.png



Gambar 296: images/PLOT3D_Diva%20Nagita_23030630024-223.png



Gambar 297: images/PLOT3D_Diva%20Nagita_23030630024-224.png



Gambar 298: images/PLOT3D_Diva%20Nagita_23030630024-225.png

> 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)'; ...

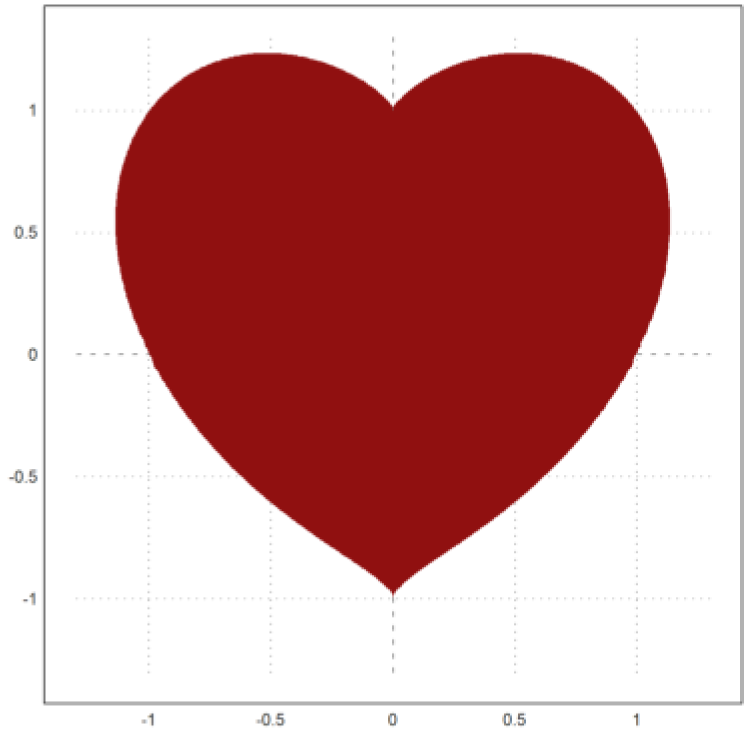
> plot3d(r*cos(t)*sin(s),r*cos(t)*cos(s),r*sin(t), ...

> >hue,<frame,color=red,zoom=4,amb=0,max=0.7,grid=12,height=50°):

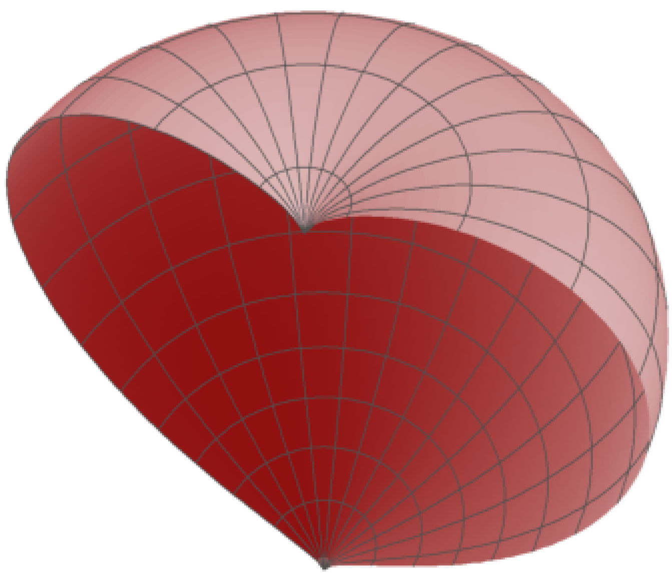
Berikut ini adalah plot 3D dari gambar di atas yang diputar di sekitar sumbu z. Kami mendefinisikan fungsi, yang menggambarkan objek.

>function f(x,y,z) ...

```
r=x^2+y^2;
```



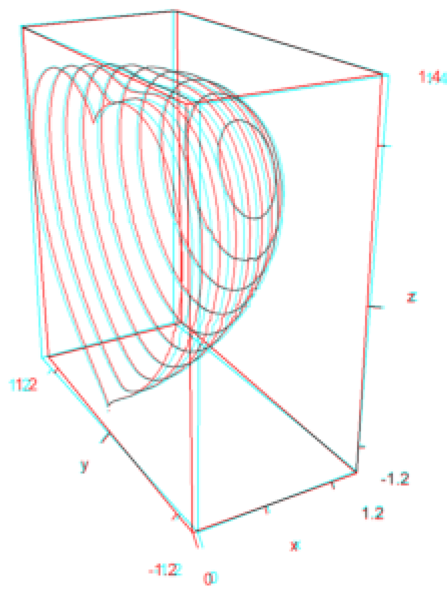
Gambar 299: images/PLOT3D_Diva%20Nagita_23030630024-226.png



Gambar 300: images/PLOT3D_Diva%20Nagita_23030630024-231.png

```
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):



Gambar 301: images/PLOT3D_Diva%20Nagita_23030630024-232.png

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:. 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
```

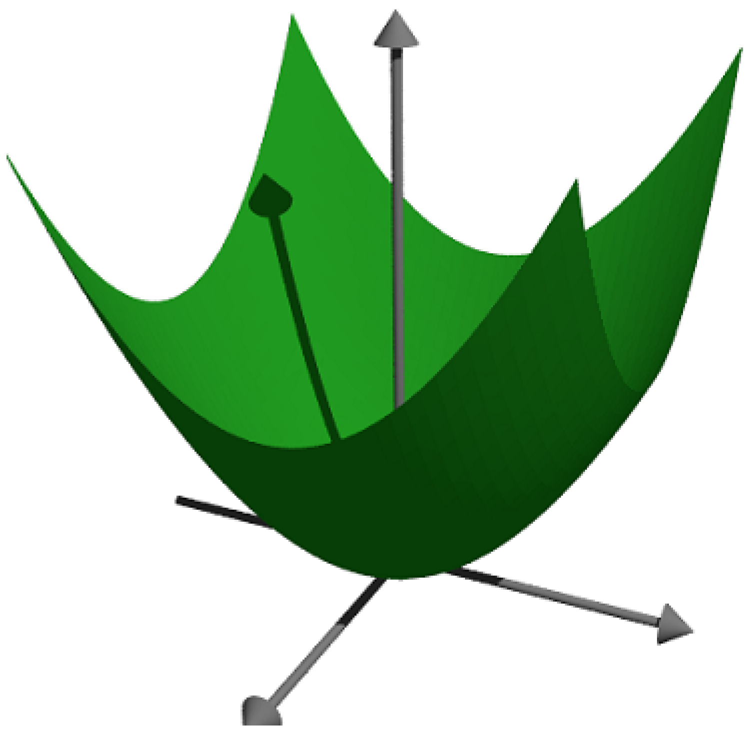
Contoh Penggunaan

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);

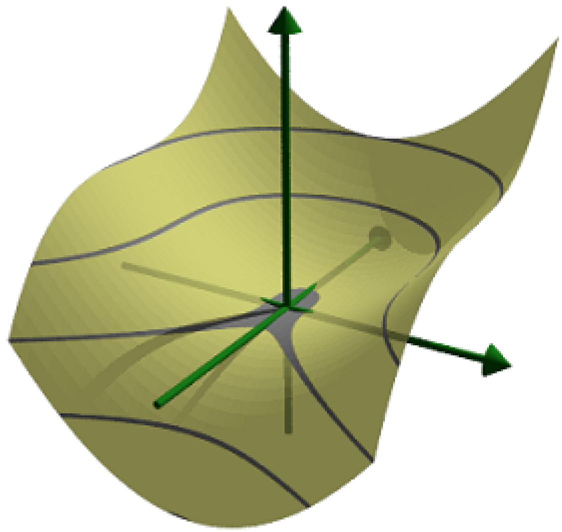


Gambar 302: images/PLOT3D_Diva%20Nagita_23030630024-233.png

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, . . .



Gambar 303: images/PLOT3D_Diva%20Nagita_23030630024-234.png

> angle=120°,level=1/40,dlevel=0.005,light=[-1,1,1],height=45°,n=50, . . .
> <fscale,zoom=3.8);

Object Povray

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

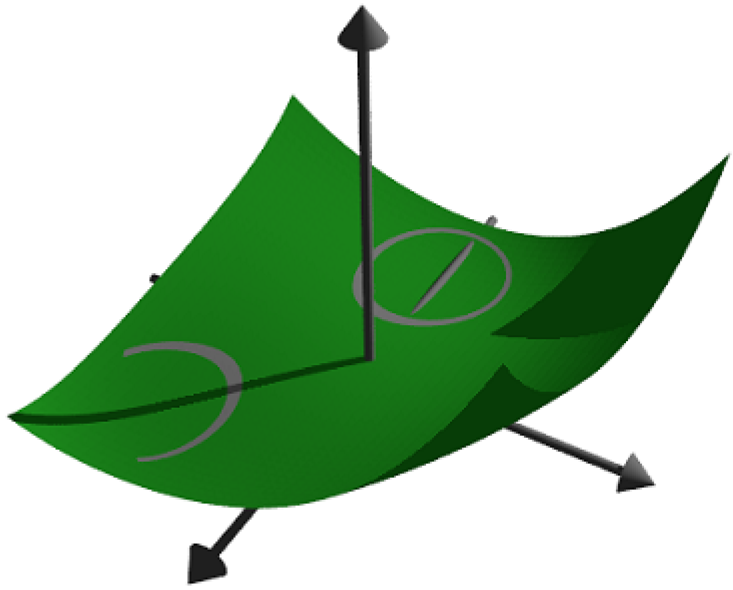
di Euler. Fungsi povx(), povy(), dll. hanya mengembalikan vektor

[1,0,0] yang dapat digunakan sebagai gantinya.

>c1

```
cylinder { &lt;-1,0,0&gt;, &lt;1,0,0&gt;, 1
 texture { pigment { color rgb &lt;0.941176,0.509804,0.392157&gt; }  }
 finish { ambient 0.2 }
 }
```

Akan ditambahkan tekstur ke objek dengan tiga warna berbeda yaitu orange, yellow, dan lightblue.



Gambar 304: images/PLOT3D_Diva%20Nagita_23030630024-235.png

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 &lt;0.101961,0.2,0.301961,0.1&gt; }  }
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 sebuah donat

>writeln(povobject(povdonat(1,0.5),povlook(lightblue,>phong),xrotate(90°)));

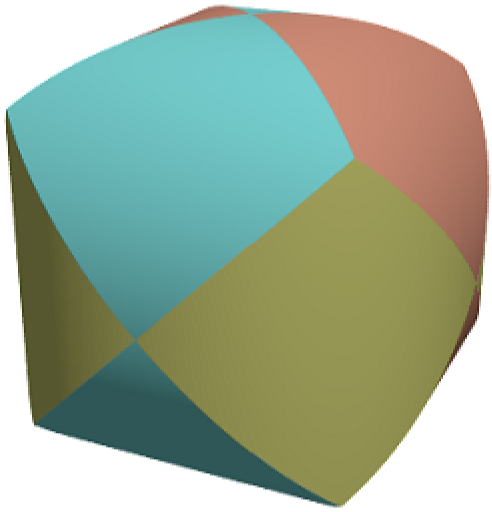
>povend();

Menggambar Grafik Tiga Dimensi alam modus anaglif

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.

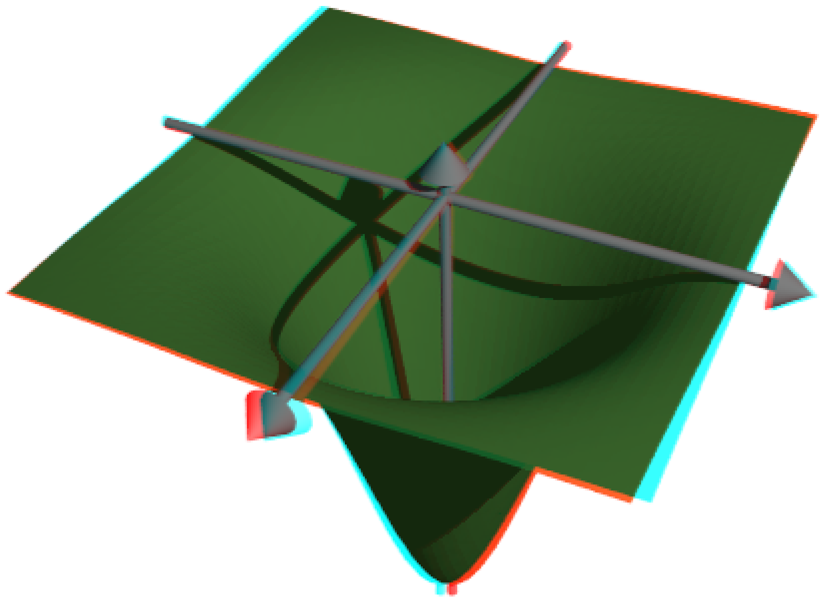
Fungsi pov3d() memiliki sakelar sederhana untuk menghasilkan anaglyphs.



Gambar 305: images/PLOT3D_Diva%20Nagita_23030630024-236.png

Gambar 306: images/PLOT3D_Diva%20Nagita_23030630024-237.png

>pov3d("-exp(-x^{2-y}2)/2",r=2,height=45°,>anaglyph, . . .
> center=[0,0,0.5],zoom=3.5);



Gambar 307: images/PLOT3D_Diva%20Nagita_23030630024-238.png

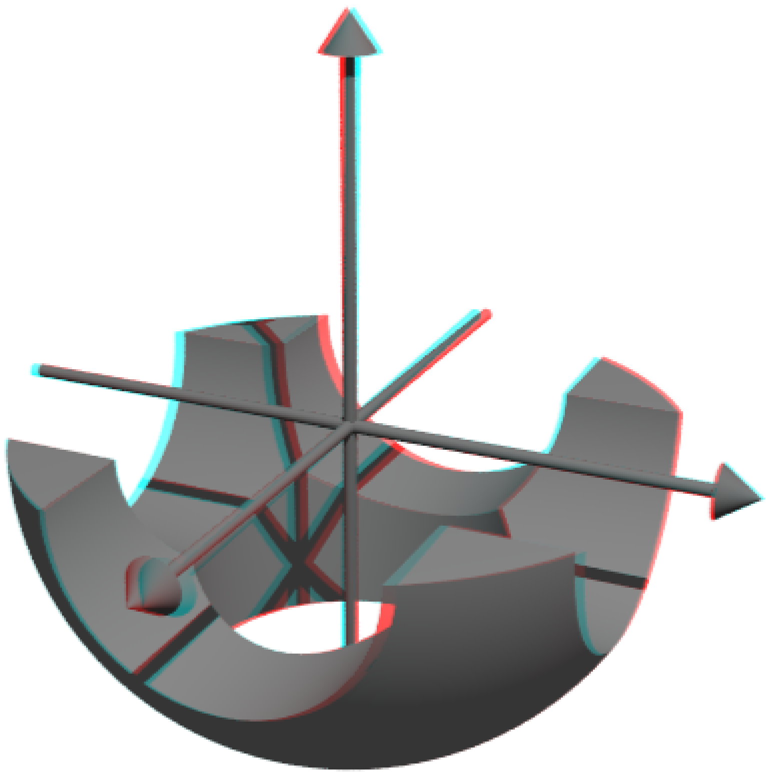
Jika Anda membuat adegan dengan objek, Anda perlu menempatkan generasi adegan ke dalam fungsi, dan menjalankannya dua kali dengan nilai yang berbeda untuk parameter anaglyph.
>function myscene . . .

```
  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);

Fungsi Implisit menggunakan Povray

Povray dapat memplot himpunan di mana f(x,y,z)=0, seperti parameter implisit di plot3d. Namun hasilnya terlihat jauh lebih baik.



Gambar 308: images/PLOT3D_Diva%20Nagita_23030630024-239.png

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();

Gambar 309: images/PLOT3D_Diva%20Nagita_23030630024-241.png

>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;

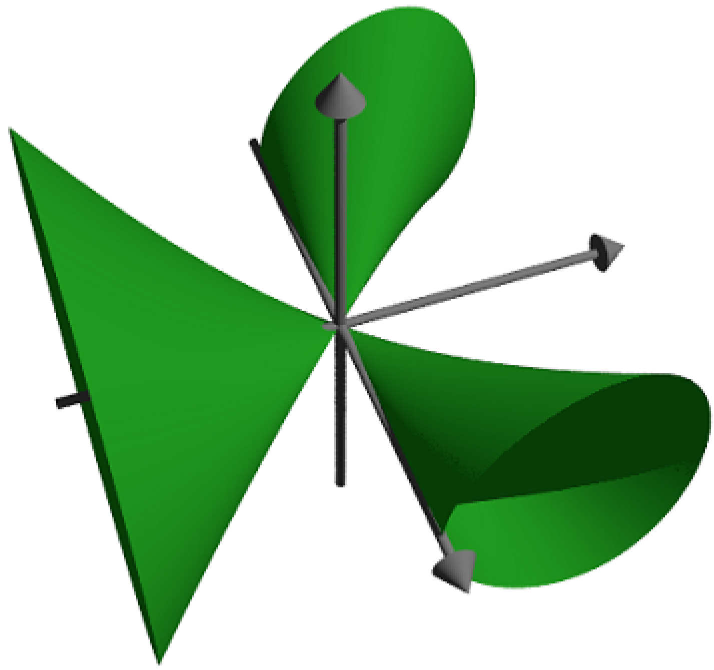
>defaultpovray="C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe"

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

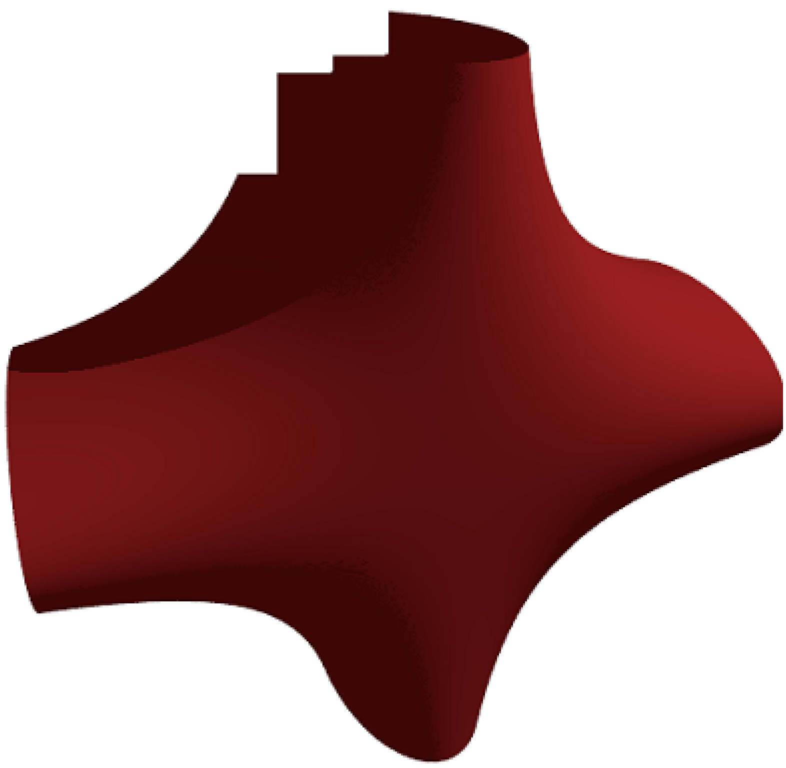
>povstart(angle=70°,height=30°);

>writeln(povsurface("pow(x,2)+pow(y,2)*pow(z,2)-1",povlook(red),povbox(-2,2,"")));

>povend();



Gambar 310: images/PLOT3D_Diva%20Nagita_23030630024-242.png



Gambar 311: images/PLOT3D_Diva%20Nagita_23030630024-243.png

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();

Gambar 312: images/PLOT3D_Diva%20Nagita_23030630024-244.png

Menggambar Titik pada ruang Tiga Dimensi (3D)

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.

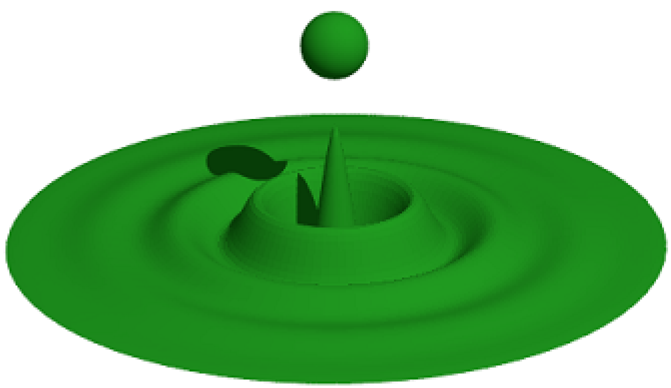
>function f(x) := x^3-x+1; ...
> x=-1:0.01:1; t=linspace(0,2pi,8)'; ...
> Z=x; X=cos(t)*f(x); Y=sin(t)*f(x); ...
> pov3d(X,Y,Z,angle=40°,height=20°,axis=0,zoom=4,light=[10,-5,5]);

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; ...

Gambar 313: images/PLOT3D_Diva%20Nagita_23030630024-245.png



Gambar 314: images/PLOT3D_Diva%20Nagita_23030630024-246.png

> 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

```
 2  3
x  y
```

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.

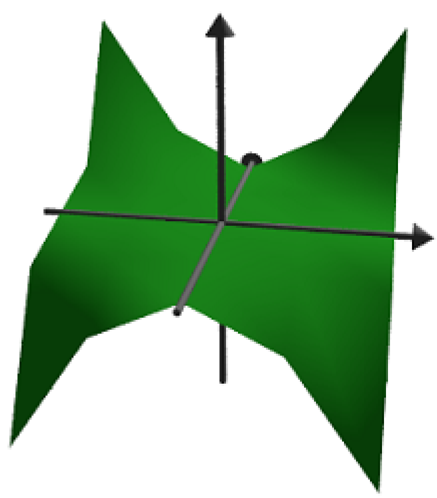
>N &= crossproduct(dx,dy); NX &= N[1]; NY &= N[2]; NZ &= N[3]; N,

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

Kami hanya menggunakan 25 poin.

>x=-1:0.5:1; y=x';

>pov3d(x,y,Z(x,y),angle=10°, ...
> xv=NX(x,y),yv=NY(x,y),zv=NZ(x,y),<shadow);



Gambar 315: images/PLOT3D_Diva%20Nagita_23030630024-247.png

Berikut ini adalah simpul Trefoil yang dilakukan oleh A. Busser di Povray. Ada versi yang ditingkatkan dari ini dalam contoh.

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.

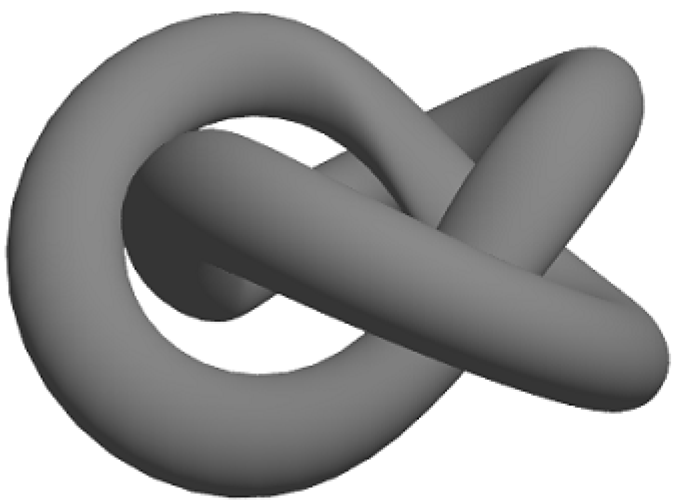
```
>dn &= crossproduct(dx,dy);
```

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));
```



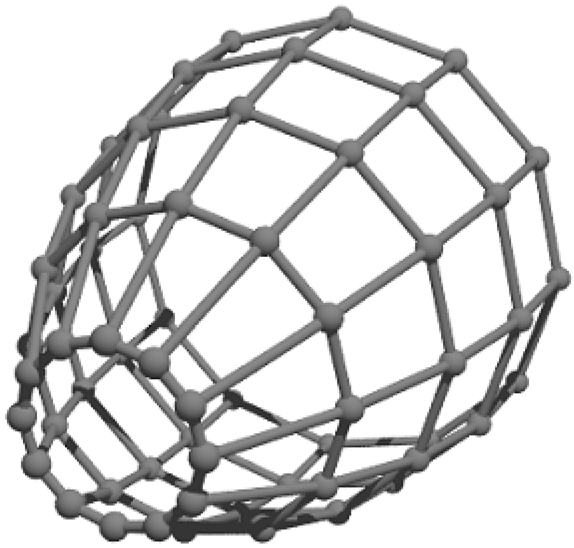
Gambar 316: images/PLOT3D_Diva%20Nagita_23030630024-248.png

Kami juga dapat menghasilkan grid dalam 3D.

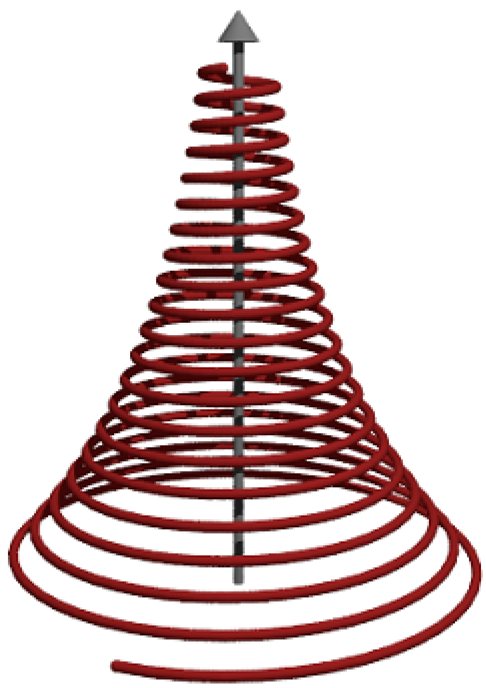
```
>povstart(zoom=4); ...
> x=-1:0.5:1; r=1-(x+1)^2/6; ...
> t=(0°:30°:360°)'; y=r*cos(t); z=r*sin(t); ...
> writeln(povgrid(x,y,z,d=0.02,dballs=0.05)); ...
> povend();
```

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();
```



Gambar 317: images/PLOT3D_Diva%20Nagita_23030630024-249.png



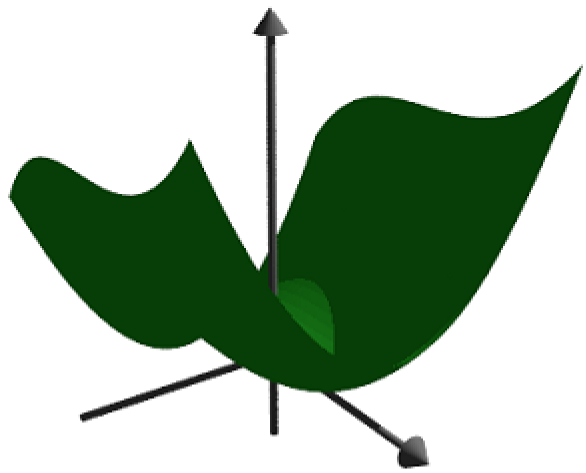
Gambar 318: images/PLOT3D_Diva%20Nagita_23030630024-250.png

Latihan Soal

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°);



Gambar 319: images/PLOT3D_Diva%20Nagita_23030630024-252.png

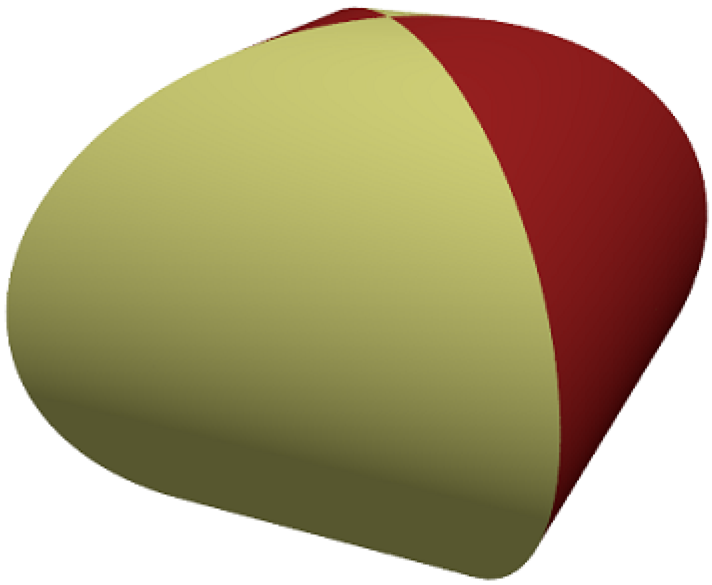
2. Buatlah gabungan 2 silinder dengan fungsi povx() berwarna merah dan povz() berwarna kuning dan zoom 4

>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):



Gambar 320: images/PLOT3D_Diva%20Nagita_23030630024-253.png

images/PLOT3D_Diva Nagita_23030630024-255.png

Gambar 321: images/PLOT3D_Diva%20Nagita_23030630024-255.png

unicodehyperref hyphensurl [ ]book xcolor amsmath,amssymb iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype [protrusion]basicmath parskip graphicx bookmark xurl same hidelinks, pdfcreator=LaTeX via pandoc

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 pemetaan setiap anggota sebuah himpunan kepada anggota himpunan yang lain. Fungsi adalah salah satu konsep dasar dari matematika dan setiap ilmu kuantitatif. Pada dasarnya, fungsi adalah suatu relasi yang memetakan setiap anggota dari suatu himpunan yang disebut sebagai daerah asal atau domain ke tepat satu anggota himpunan lain yang disebut daerah kawan atau kodomain.

Adapun beberapa jenis fungsi yaitu:

Fungsi Aljabar

Fungsi aljabar adalah fungsi yang dapat didefinisikan sebagai akar dari sebuah persamaan aljabar. Fungsi aljabar merupakan ekspresi aljabar menggunakan sejumlah suku terbatas, yang melibatkan operasi aljabar seperti penambahan, pengurangan, perkalian, pembagian, dan peningkatan menjadi pangkat pecahan.

Sifat-Sifat Fungsi Aljabar

1. Fungsi Injektif: Fungsi aljabar dapat menjadi injektif jika setiap elemen di domain dipetakan ke elemen yang berbeda di kodomain. Artinya, jika

$$f(a1) = f(a2)$$

maka

$$a1 = a2$$

2. Fungsi Surjektif: Fungsi aljabar dapat menjadi surjektif jika setiap elemen di kodomain memiliki setidaknya satu elemen di domain yang memetakan ke sana. Artinya, untuk setiap

b di kodomain, ada a di domain sehingga

$$f(a) = b$$

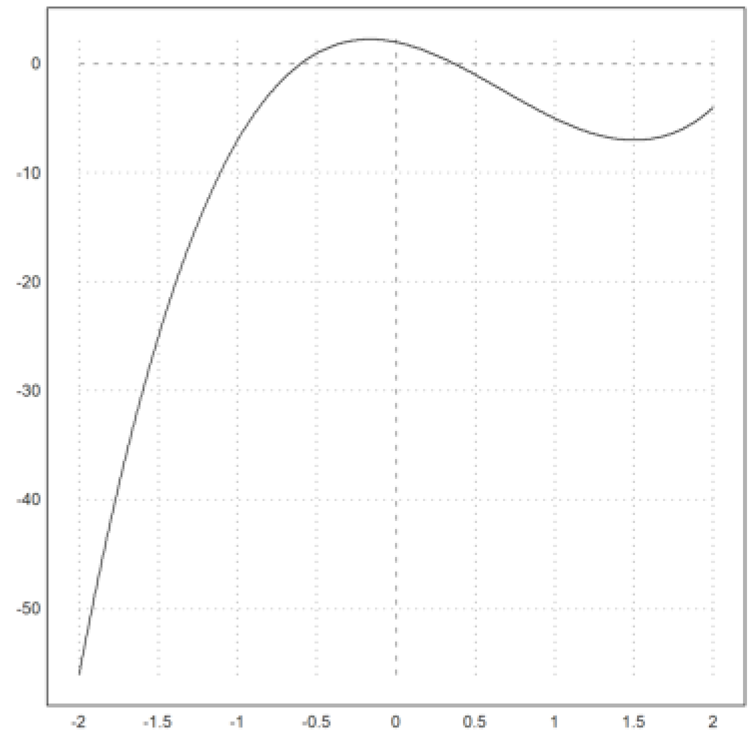
3. Fungsi Bijektif: Fungsi aljabar yang juga bijektif harus memenuhi keduanya, yaitu injektif dan surjektif. Artinya, setiap elemen di domain dipetakan ke elemen unik di kodomain, dan setiap elemen di kodomain dipetakan oleh elemen di domain

Fungsi aljabar dapat memiliki berbagai bentuk kurva tergantung pada jenisnya. Berikut beberapa contoh:

1. Fungsi Polinomial

$$y = 4x^3 - 8x^2 - 3x + 2$$

>plot2d(“4*x^{3-8*x}2-3*x+2”):



Gambar 322: images/KALKULUS_Diva%20Nagita(23030630024)-005.png

2. Fungsi Rasional

$$f(x) = \frac{x^2 + 1}{x - 1}$$

>reset;
>plot2d(“(x^2+1)/(x-1)”):

3. Fungsi Radikal

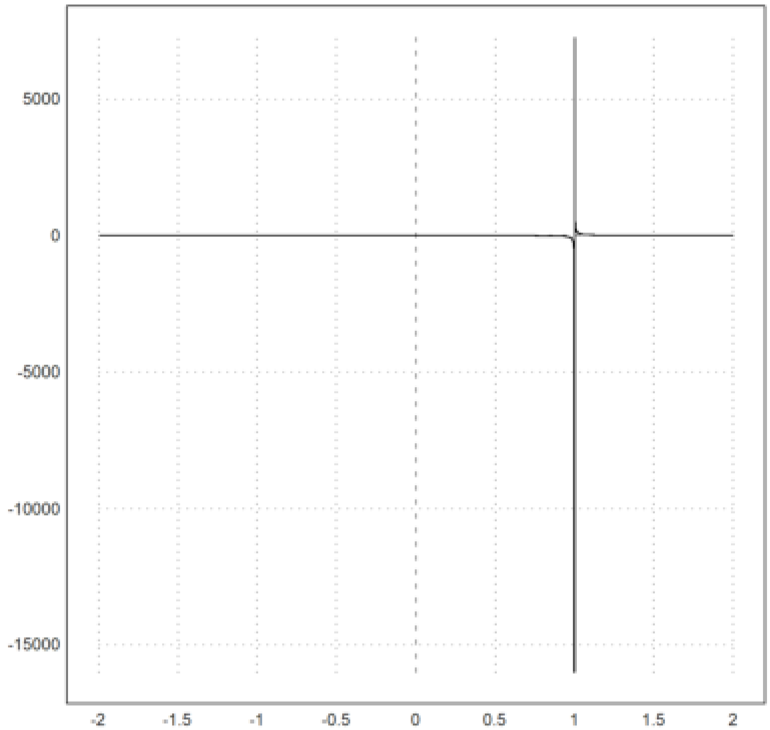
$$\sqrt{x}$$

>reset;
>plot2d(“sqrt(x)”):

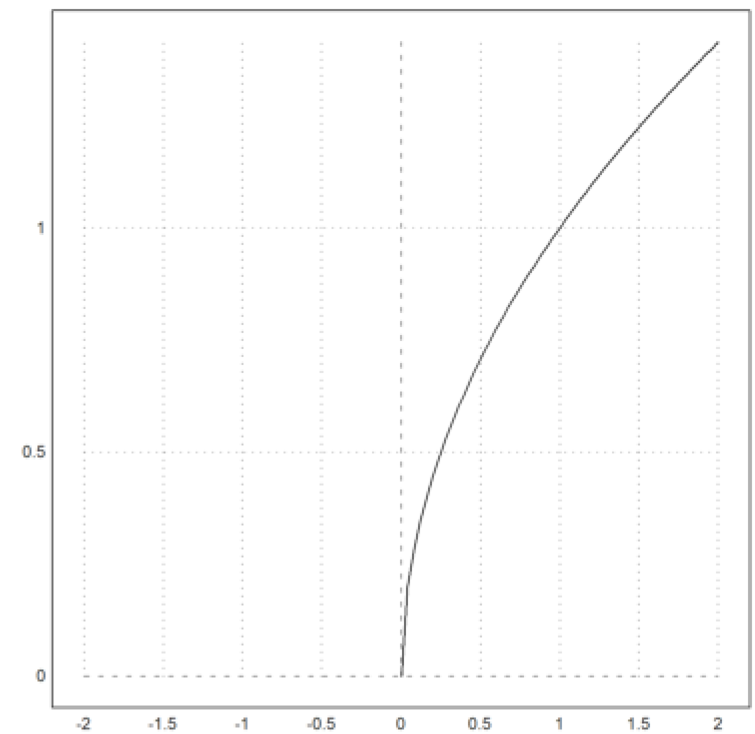
Fungsi Trigonometri

Fungsi trigonometri adalah fungsi matematika yang mengaitkan sudut dari segitiga siku-siku dengan perbandingan antara dua sisi segitiga, serta dapat didefinisikan melalui lingkaran satuan. Fungsi ini mencakup enam jenis utama: sinus (sin), kosinus (cos), tangen (tan), kosekan (csc), sekan (sec), dan kotangen (cot).

Berikut adalah beberapa sifat utama fungsi trigonometri,



Gambar 323: images/KALKULUS_Diva%20Nagita(23030630024)-007.png



Gambar 324: images/KALKULUS_Diva%20Nagita(23030630024)-009.png

- Periodisitas: Fungsi sinus dan kosinus memiliki periode 2pi, sedangkan tangen memiliki periode pi.
- Amplitudo: Jarak dari garis tengah ke titik maksimum atau minimum; untuk sinus dan kosinus, amplitudonya adalah 1.
- Nilai Maksimum dan Minimum: Sinus dan kosinus memiliki nilai maksimum +1 dan minimum -1; tangen tidak terbatas.

Titik Asimtot: Terdapat pada fungsi tangen di

$$\frac{\pi}{2} + k\pi$$

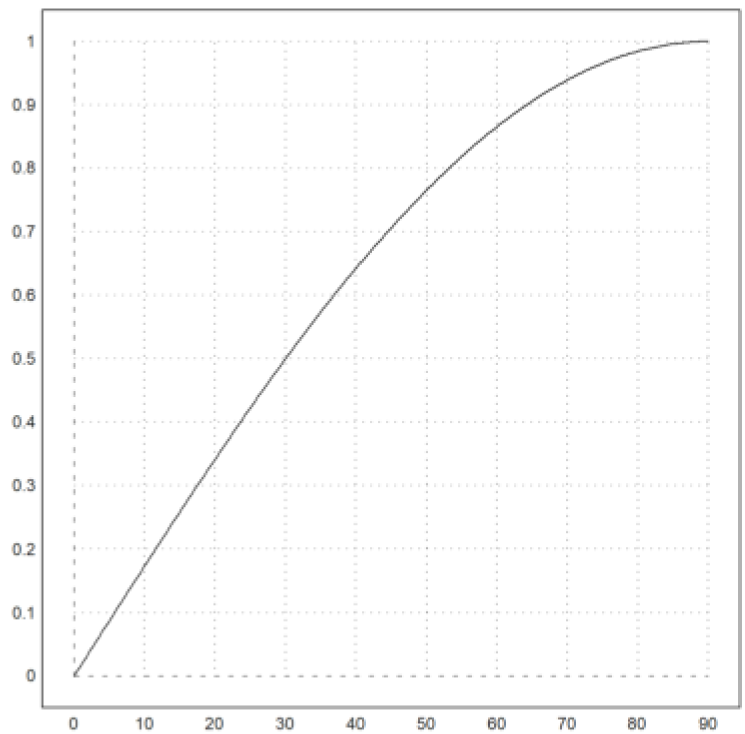
untuk setiap bilangan bulat k.
Fungsi trigonometri memiliki berbagai bentuk kurva tergantung pada jenisnya. Berikut beberapa contoh:

1. Sinus

>sin(90°)

```
1
```

>plot2d("sin(x*pi/180)", 0, 90):



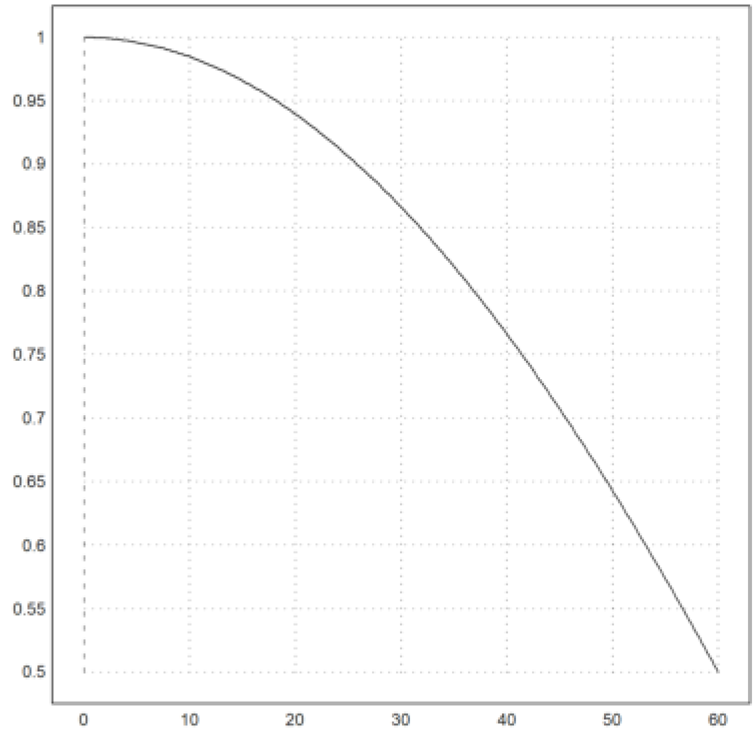
Gambar 325: images/KALKULUS_Diva%20Nagita(23030630024)-011.png

2. Kosinus

>cos(60)

```
-0.952412980415
```

>plot2d("cos(x*pi/180)", 0, 60):



Gambar 326: images/KALKULUS_Diva%20Nagita(23030630024)-012.png

3. Tangen

>tan(45°)

```
1
```

>plot2d("tan(x*pi/180)", 0, 45):

Fungsi Eksponensial

Fungsi eksponensial adalah fungsi matematika yang memiliki bentuk umum:

$$f(x) = a^x$$

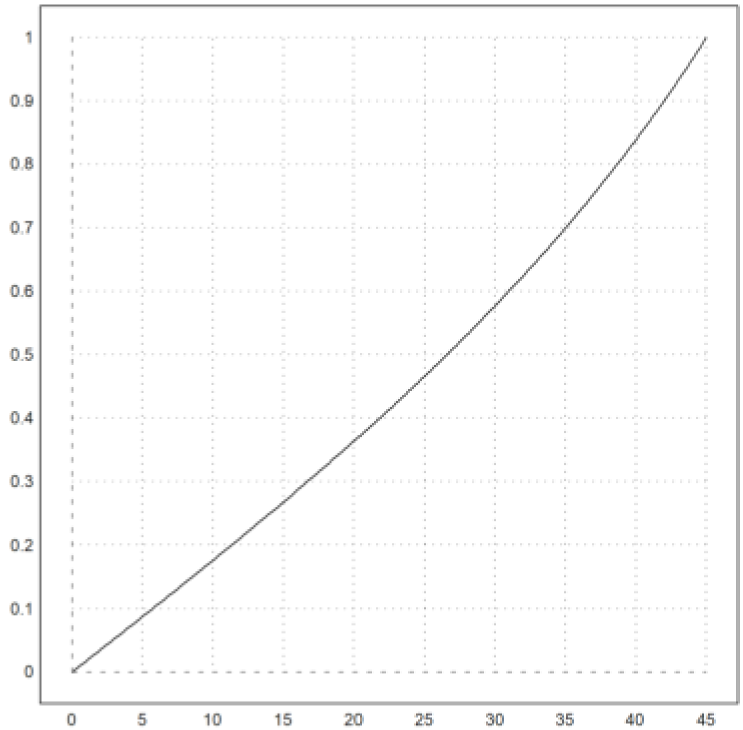
dimana a adalah bilangan positif yang disebut basis (biasanya), dan x adalah eksponen.
Adapun sifat-sifat dari fungsi eksponensial adalah

- Untuk semua nilai x, fungsi eksponensial selalu menghasilkan nilai positif, yaitu f(x)>0 untuk setiap x.
- Jika a>1, fungsi eksponensial akan meningkat.
- Jika 0<a<1, fungsi eksponensial akan menurun.
- Setiap bilangan yang dipangkatkan nol sama dengan satu, yaitu

$$a^0 = 1$$

* Setiap bilangan yang dipangkatkan satu sama dengan bilangan itu * sendiri, yaitu * aˆ1=a

$$a^1 = a$$



Gambar 327: images/KALKULUS_Diva%20Nagita(23030630024)-013.png

* Untuk basis yang sama, berlaku sifat penjumlahan: * aˆm . a$^{n=a}${m+n}

$$a^m.a^n = a^{m+n}$$

* Untuk pembagian dengan basis yang sama, berlaku sifat pengurangan: * a$^m \frac{}{a^n = a\{m-n\}}$

$$\frac{a^m}{a^n} = a^{m-n}$$

* Grafik fungsi eksponensial mendekati sumbu x (y = 0) tetapi tidak * pernah menyentuhnya, sehingga memiliki asimtot horizontal di y = 0.

  - Turunan dari fungsi eksponensial dengan basis e adalah unik karena
  - hasilnya adalah fungsi itu sendiri, yaitu

- d$\frac{}{dx e^x = e^x}$

$$\frac{d}{dx}e^x = e^x$$

Contoh Kurva fungsi eksponensial:

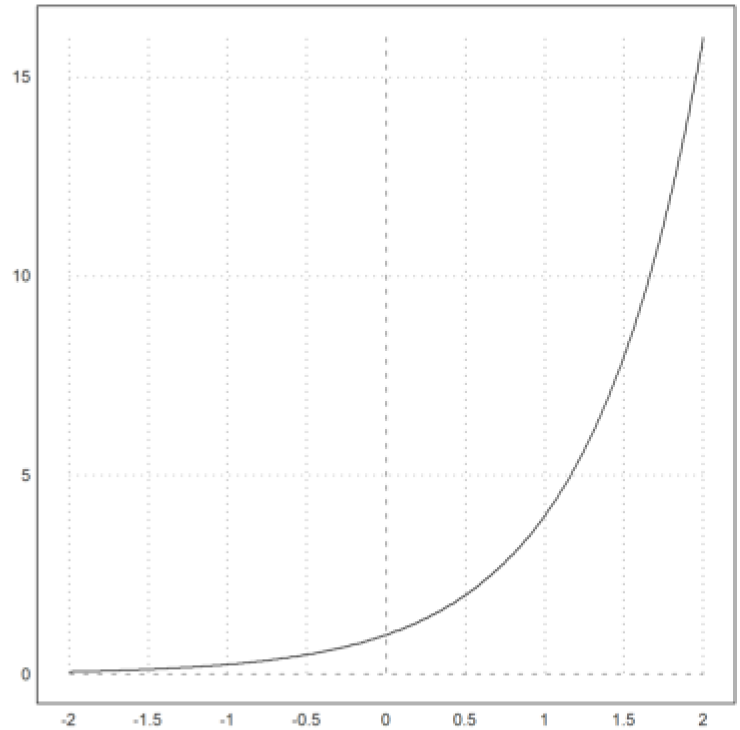
$$4^x$$

>plot2d("4ˆx", -2, 2):

Fungsi Logaritma

Fungsi logaritma adalah fungsi matematis yang merupakan invers dari fungsi eksponensial. Secara formal, jika kita memiliki fungsi eksponensial

$$f(x) = a^x$$



Gambar 328: images/KALKULUS_Diva%20Nagita(23030630024)-021.png

(dengan a>0 dan atidak sama dengan 1), maka fungsi logaritma dapat dinyatakan sebagai

$$f^{-1}(x) = log_a(x)$$

yang berarti logaritma dari x dengan basis a adalah eksponen yang harus dipangkatkan pada a untuk mendapatkan x. Fungsi logaritma memiliki bentuk umum

$$f(x) = alog_b(x)$$

dengan syarat x>0.
Fungsi logaritma memiliki beberapa sifat penting:

1. Sifat Penjumlahan:

$$log_a(m.n) = log_a(m) + log_a(n)$$

2. Sifat Pengurangan:

$$log_a(\frac{m}{n}) = log_a(m) - log_a(n)$$

3. Sifat Pangkat:

$$log_a(m^p) = p.log_a(m)$$

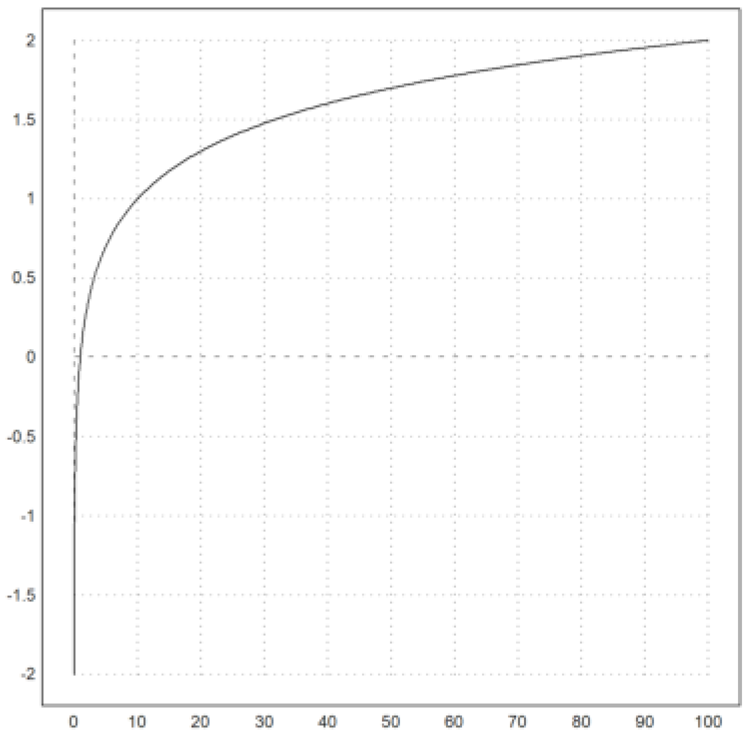
4. Perubahan Basis:

$$log_a(b) = \frac{log_c(b)}{log_c(a)}$$

untuk basis yang berbeda.

Berikut contoh kurva fungsi logaritma sederhana:

>plot2d("log10(x)", 0.01, 100):



Gambar 329: images/KALKULUS_Diva%20Nagita(23030630024)-029.png

>plot2d("ln(x)", "log10(x)", 1, 10):

Komposisi Fungsi

Komposisi fungsi adalah operasi yang menggabungkan dua atau lebih fungsi untuk membentuk fungsi baru. Dalam notasi, jika terdapat dua fungsi f dan g, komposisi fungsi ditulis sebagai (f o g)(x),

yang berarti kita pertama-tama menerapkan fungsi x, kemudian hasilnya digunakan sebagai input untuk fungsi f. Secara matematis, ini dapat dinyatakan sebagai (f o g)(x)=f(g(x)).

Adapun sifat-sifat komposisi fungsi

1. Asosiatif:

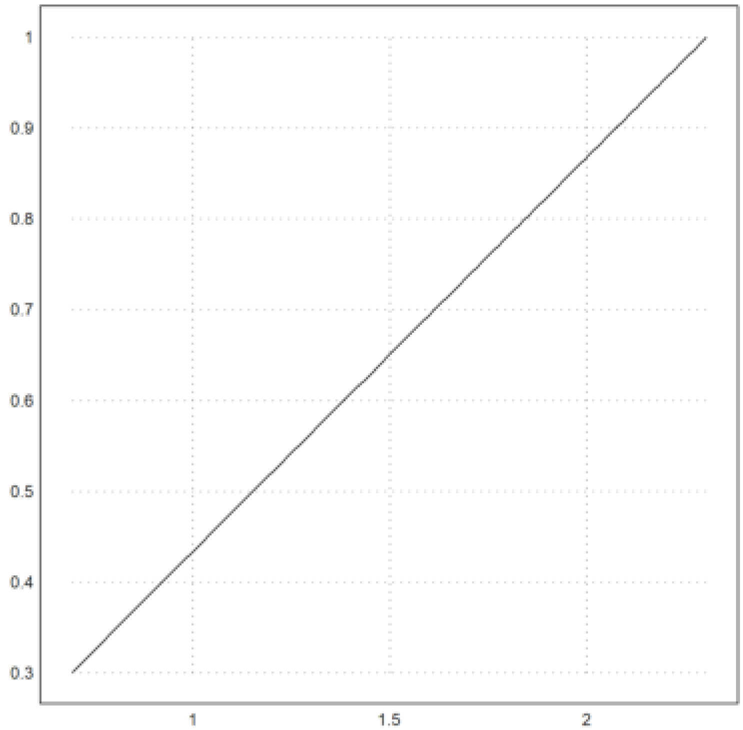
$$(fo(goh))(x) = ((fog)oh)(x)$$

untuk semua fungsi f,g,h yang terdefinisi dengan baik.

2. Identitas:

$$(foI)(x) = I(x)$$

dimana I(x) adalah fungsi identitas yang memetakan setiap elemen ke dirinya sendiri.



Gambar 330: images/KALKULUS_Diva%20Nagita(23030630024)-030.png

3. Tidak komutatif:

$$(fog)(x)$$

tidak sama dengan

$$(goh)(x)$$

kecuali dalam kasus tertentu di mana kedua fungsi tersebut saling berkomutasi.

4. Bijektif: Jika dua fungsi yang dikomposisikan adalah bijektif, maka hasil komposisinya juga bijektif. Ini berarti bahwa jika

f dan g masing-masing injektif dan surjektif, maka f o g juga memiliki sifat ini.
Berikut contoh kurva komposisi fungsi:

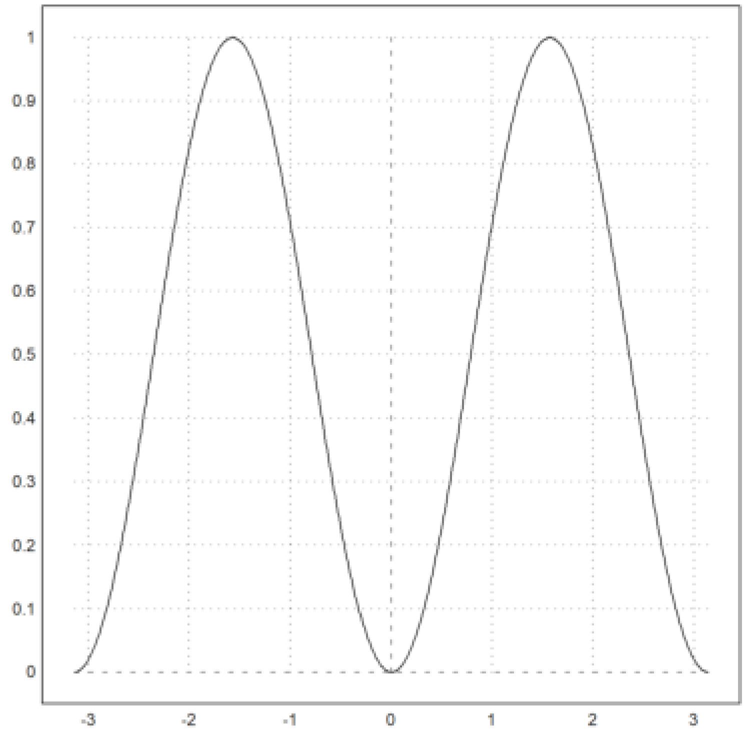
1.

$$f(x) = x^2$$

$$g(x) = sinx$$

$$h(x) = f(g(x))$$

>function f(x) := xˆ2
>function g(x) := sin(x)
>function h(x) := f(g(x))
>plot2d("h(x)", -pi, pi):



Gambar 331: images/KALKULUS_Diva%20Nagita(23030630024)-038.png

2.

$$f(x) = cosx$$
$$g(x) = x^3 - 1$$
$$f(g(x))$$

>function f(x) := cos(x)
>function g(x) := x^3 - 1
>function h(x) := f(g(x))
>plot2d("h(x)", -1, 2):

Limit Fungsi

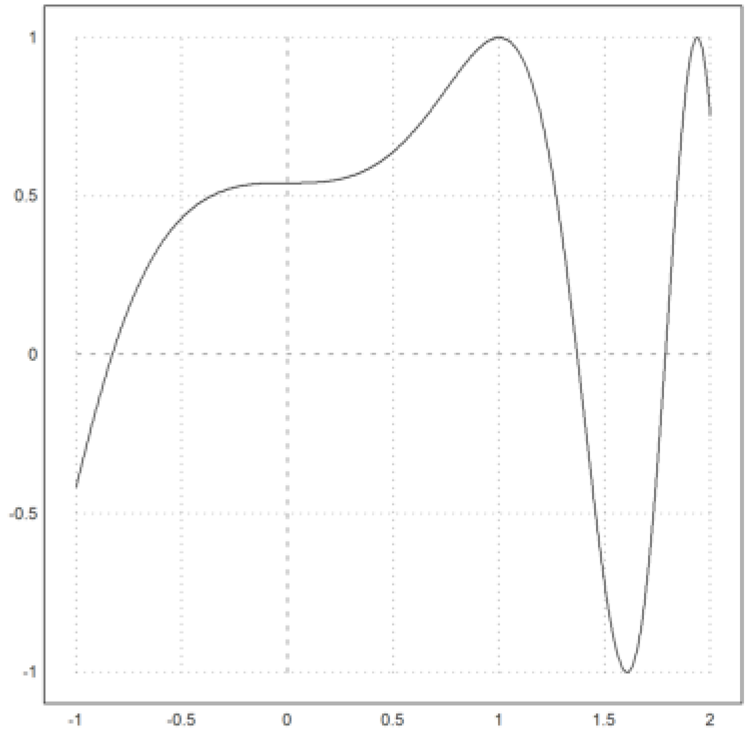
Limit adalah konsep dasar dalam kalkulus yang menggambarkan perilaku suatu fungsi saat mendekati nilai tertentu. Limit digunakan untuk memahami perubahan nilai fungsi ketika variabel mendekati suatu titik. Limit dari suatu fungsi f(x)pada titik c adalah nilai yang didekati oleh f(x) saat mendekati c. Notasi limit dinyatakan sebagai:

$$\lim_{x \to c} f(x) = L$$

1. Limit Kiri dan Kanan

- Limit kiri: limit dari fungsi(x) saat x mendekati c dari sisi kiri atau nilai x lebih kecil dari c dilambangkan dengan

$$\lim_{x \to c^-} f(x) = L^-$$



Gambar 332: images/KALKULUS_Diva%20Nagita(23030630024)-042.png

* Limit kanan: limit dari fungsi(x) saat x mendekati c dari sisi kanan * atau nilai x lebih besar dari c dilambangkan dengan * $\lim$ _{$x \to c$^{+}f(x)=L}$+$

$$\lim_{x \to c^+} f(x) = L^+$$

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).

Berikut visualisasi limit ke :

1. Fungsi Aljabar

$$\lim_{x \to 1} \frac{x^2 - 1}{x - 1} = 2$$

2. Fungsi Trigonometri

$$\lim_{x \to 0} \frac{\sin(x)}{x} = 1$$

3. Fungsi Eksponensial

$$\lim_{x \to -\infty} 2^x = 0$$

4. Fungsi Logaritma

$$\lim_{x \to 0^+} \log(x) = -\infty$$

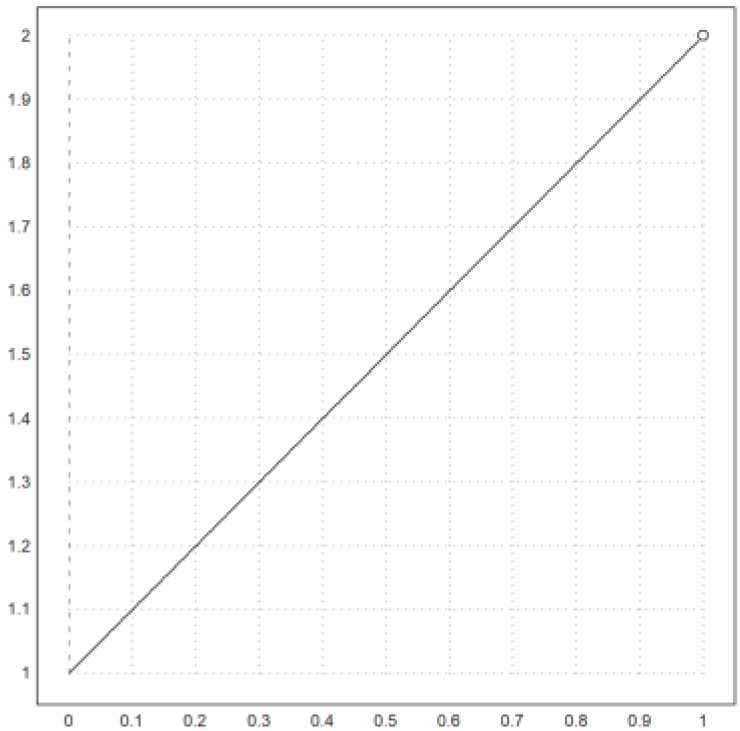
5. Komposisi Fungsi

$$\lim_{x\to\frac{\pi}{2}} (g(x))^2 = 1$$

>$limit((x^2-1)/(x-1),x,1)

$$2$$

>function f(x):= x^2-1/x-1
>aspect(1.0); plot2d("(x^2-1)/(x-1)",0,1); plot2d(1,2,>points,style="ow",>add):



Gambar 333: images/KALKULUS_Diva%20Nagita(23030630024)-052.png

>$limit((sin (x))/(x),x,0)

$$1$$

>function f(x):= sin(x)/x
>aspect(1.5); plot2d("(sin(x))/(x)",0,1); plot2d(0,1,>points,style="ow",>add):
>$showev('limit(sqrt(x^2-3*x)/(x+1),x,inf))

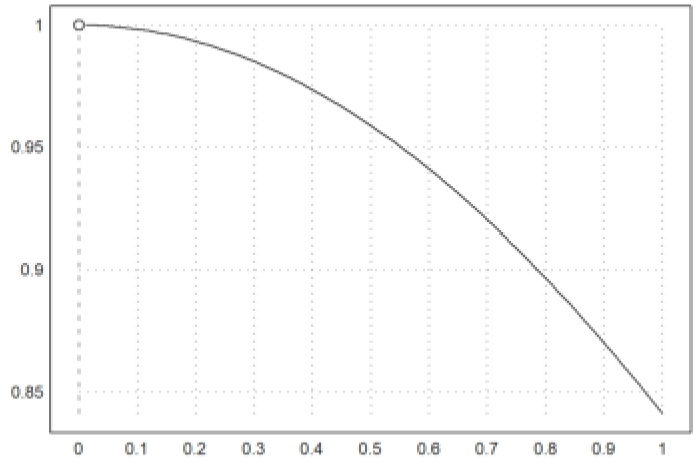
$$\lim_{x\to\infty} \frac{\sqrt{-3\,x+x^2}}{1+x} = 1$$

>$limit(($x^{3-13*x}$2+51*x-63)/($x^{3-4*x}$2-3*x+18),x,3)

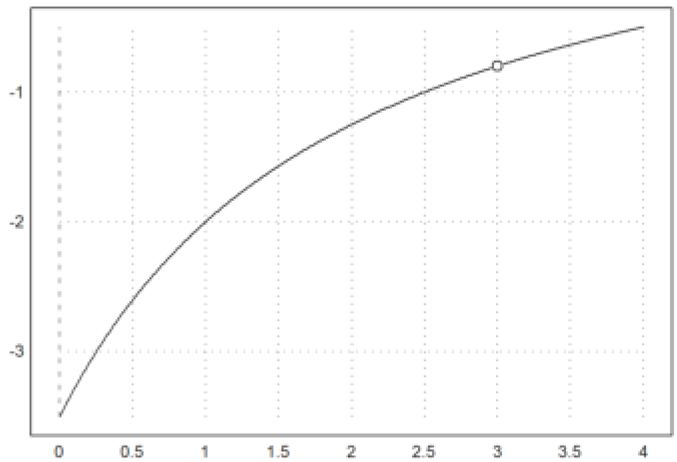
$$-\frac{4}{5}$$

>aspect(1.5); plot2d("(x^{3-13*x}2+51*x-63)/(x^{3-4*x}2-3*x+18)",0,4); plot2d(3,-4/5,>points,style="ow",>add):
>$limit(2*x*sin(x)/(1-cos(x)),x,0)

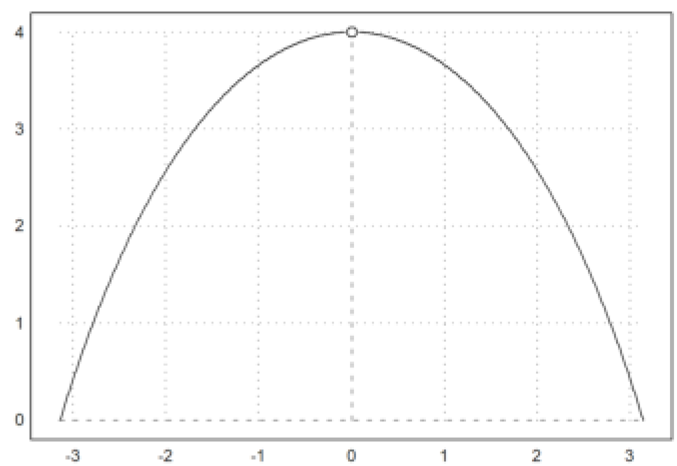
$$4$$



Gambar 334: images/KALKULUS_Diva%20Nagita(23030630024)-054.png



Gambar 335: images/KALKULUS_Diva%20Nagita(23030630024)-057.png

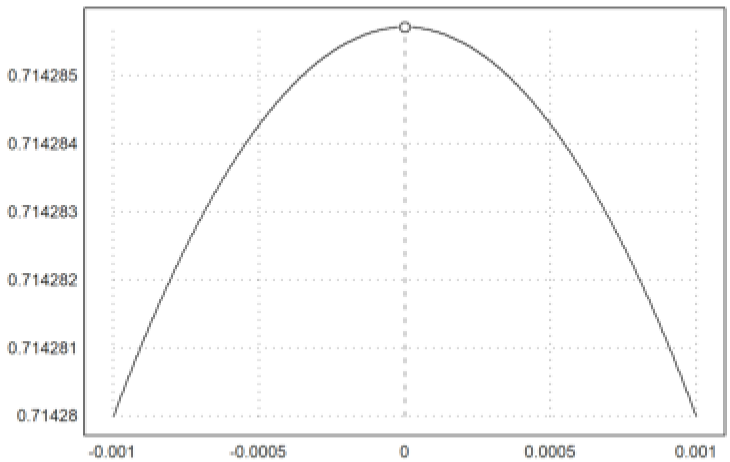


Gambar 336: images/KALKULUS_Diva%20Nagita(23030630024)-059.png

>plot2d("2*x*sin(x)/(1-cos(x))",-pi,pi); plot2d(0,4,>points,style="ow",>add):
>$limit(cot(7*h)/cot(5*h),h,0)

$$\frac{5}{7}$$

>plot2d("cot(7*x)/cot(5*x)",-0.001,0.001); plot2d(0,5/7,>points,style="ow",>add):



Gambar 337: images/KALKULUS_Diva%20Nagita(23030630024)-061.png

>$showev('limit(((x/8)^(1/3)-1)/(x-8),x,8))

$$\lim_{x\rightarrow 8}{\frac{-1+\frac{x^{\frac{1}{3}}}{2}}{-8+x}}=\frac{1}{24}$$

>$showev('limit(1/(2*x-1),x,0))

$$\lim_{x\rightarrow 0}{\frac{1}{-1+2\,x}}=-1$$

>$showev('limit((x^2-3*x-10)/(x-5),x,5))

$$\lim_{x\rightarrow 5}{\frac{-10-3\,x+x^2}{-5+x}}=7$$

>$showev('limit(sqrt(x^2+x)-x,x,inf))

$$\lim_{x\rightarrow \infty }{-x+\sqrt{x+x^2}}=\frac{1}{2}$$

>$showev('limit(abs(x-1)/(x-1),x,1,minus))

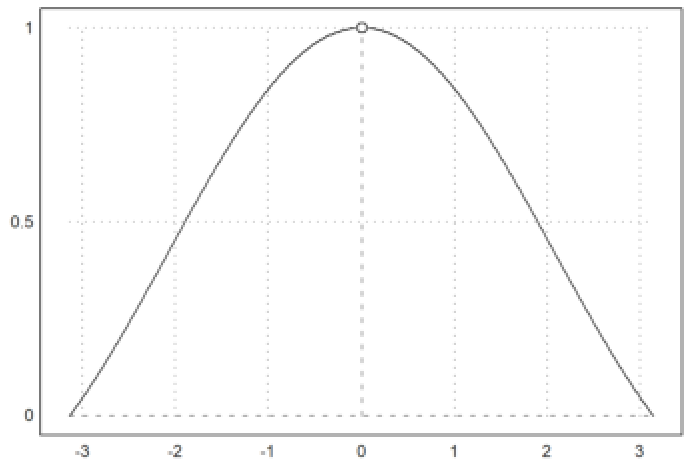
$$\lim_{x\uparrow 1}{\frac{\left| -1+x\right| }{-1+x}}=-1$$

>$showev('limit(sin(x)/x,x,0))

$$\lim_{x\rightarrow 0}{\frac{\sin x}{x}}=1$$

>plot2d("sin(x)/x",-pi,pi); plot2d(0,1,>points,style="ow",>add):
>$showev('limit(sin(x^3)/x,x,0))

$$\lim_{x\rightarrow 0}{\frac{\sin x^3}{x}}=0$$



Gambar 338: images/KALKULUS_Diva%20Nagita(23030630024)-068.png

>$showev('limit(log(x), x, minf))

$$\lim_{x\rightarrow -\infty }{\log x}=\mathit{infinity}$$

>$showev('limit((-2)^x,x, inf))

$$\lim_{x\rightarrow \infty }{\left(-2\right)^{x}}=\mathit{infinity}$$

>$showev('limit(t-sqrt(2-t),t,2,minus))

$$\lim_{t\uparrow 2}{-\sqrt{2-t}+t}=2$$

>$showev('limit(t-sqrt(2-t),t,2,plus))

$$\lim_{t\downarrow 2}{-\sqrt{2-t}+t}=2$$

>$showev('limit(t-sqrt(2-t),t,5,plus)) // Perhatikan hasilnya

$$\lim_{t\downarrow 5}{-\sqrt{2-t}+t}=5-\sqrt{3}\,i$$

>plot2d("x-sqrt(2-x)",0,2):
>$showev('limit((x$^{2-9)/(2*x}$2-5*x-3),x,3))

$$\lim_{x\rightarrow 3}{\frac{-9+x^2}{-3-5\,x+2\,x^2}}=\frac{6}{7}$$

>$showev('limit((1-cos(x))/x,x,0))

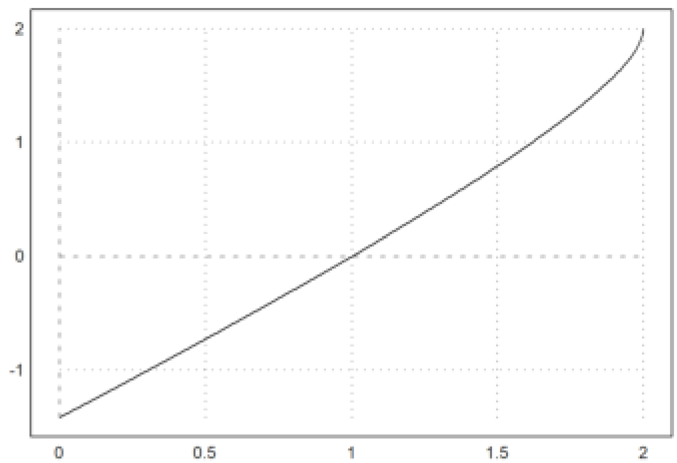
$$\lim_{x\rightarrow 0}{\frac{1-\cos x}{x}}=0$$

>$showev('limit((x$^{2+abs(x))/(x}$2-abs(x)),x,0))

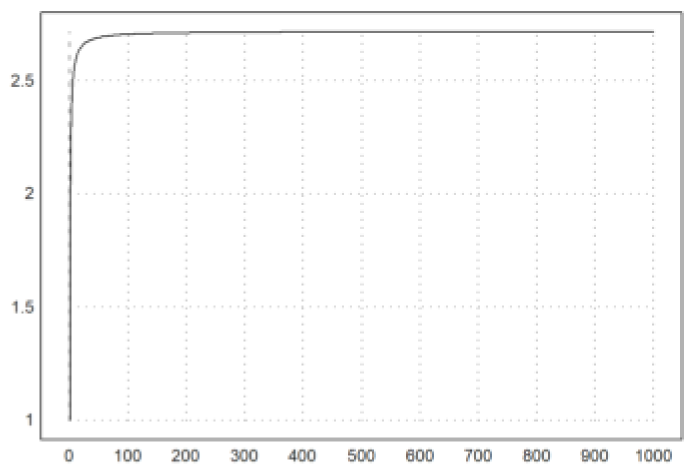
$$\lim_{x\rightarrow 0}{\frac{x^2+\left| x\right| }{x^2-\left| x\right| }}=-1$$

>$showev('limit((1+1/x)^x,x,inf))

$$\lim_{x\rightarrow \infty }{\left(1+\frac{1}{x}\right)^{x}}=e$$



Gambar 339: images/KALKULUS_Diva%20Nagita(23030630024)-075.png



Gambar 340: images/KALKULUS_Diva%20Nagita(23030630024)-080.png

>plot2d("(1+1/x)^x",0,1000):
>$showev('limit((1+k/x)^x,x,inf))

$$\lim_{x\to\infty}\left(1+\frac{1}{r\,x}\right)^{x}=e^{\frac{1}{r}}$$

>$showev('limit((1+x)^(1/x),x,0))

$$\lim_{x\to 0}\left(1+x\right)^{\frac{1}{x}}=e$$

>$showev('limit((x/(x+k))^x,x,inf))

$$\lim_{x\to\infty}\left(\frac{x}{\frac{1}{r}+x}\right)^{x}=e^{-\frac{1}{r}}$$

>$showev('limit((E$^{x\text{-}E}$2)/(x-2),x,2))

$$\lim_{x\to 2}\frac{-e^{2}+e^{x}}{-2+x}=e^{2}$$

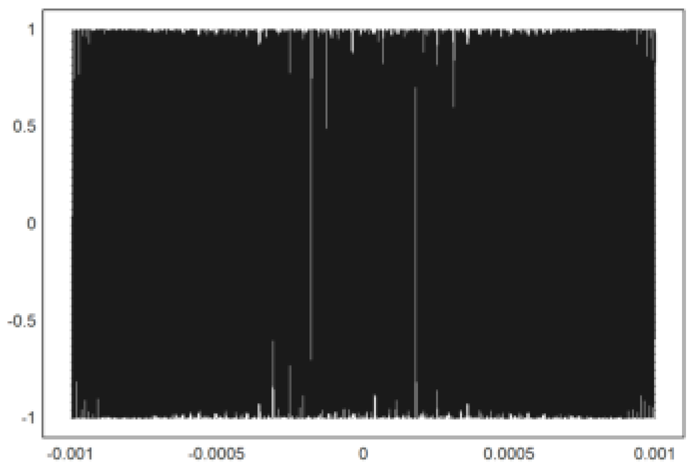
>$showev('limit(sin(1/x),x,0))

$$\lim_{x\to 0}\sin\left(\frac{1}{x}\right)=ind$$

>$showev('limit(sin(1/x),x,inf))

$$\lim_{x\to\infty}\sin\left(\frac{1}{x}\right)=0$$

>plot2d("sin(1/x)",-0.001,0.001):



Gambar 341: images/KALKULUS_Diva%20Nagita(23030630024)-087.png

Limit Kanan

Seperti yang sudah diketahui bahwa bentuk limit kanan adalah

$$\lim_{x\to c^{+}}f(x)=L^{+}$$

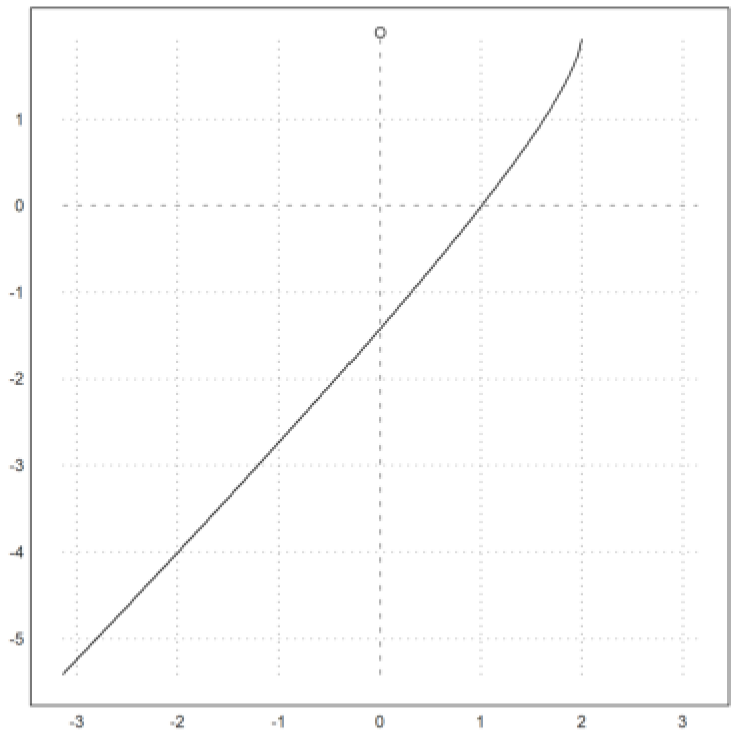
merupakan nilai yang didekati oleh f(x) saat x mendekati a dari arah kanan (atau dari yang lebih besar dari a). Contohnya:

$$\lim_{x\to 2^+} \frac{1}{x}$$

>reset;
>$showev('limit(x-sqrt(2-x),x,2,plus))

$$\lim_{x\downarrow 2} -\sqrt{2-x}+x=2$$

>plot2d("x-sqrt(2-x)",-pi,pi); plot2d(0,2,>points,style="ow",>add):



Gambar 342: images/KALKULUS_Diva%20Nagita(23030630024)-091.png

Limit Kiri

Seperti yang sudah diketahui bahwa bentuk limit kanan adalah

$$\lim_{x\to c^-} f(x) = L^-$$

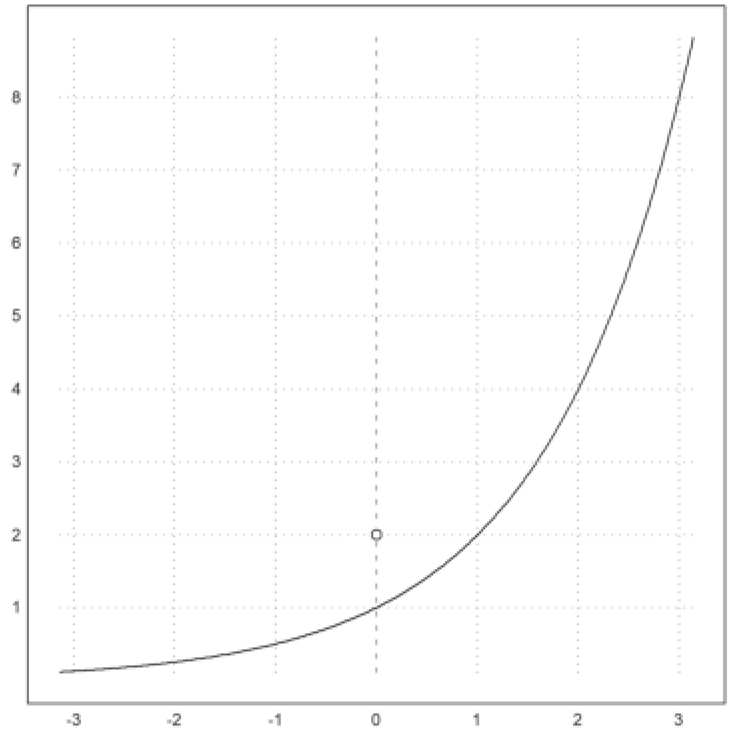
merupakan nilai yang didekati oleh f(x) saat x mendekati a dari arah kiri (atau dari yang lebih kecil dari a). Contoh:

$$\lim_{x\to 1^-} 2^x$$

>reset;
>$showev('limit(abs(2^x),x,2,minus))

$$\lim_{x\uparrow 2} 2^x = 4$$

>plot2d("2^x",-pi,pi); plot2d(0,2,>points,style="ow",>add):



Gambar 343: images/KALKULUS_Diva%20Nagita(23030630024)-095.png

Turunan Fungsi

Turunan merupakan salah satu materi lanjutan dari limit fungsi.

Turunan dapat disebut juga sebagai diferensial dan proses dalam menentukan turunan suatu fungsi disebut sebagai diferensiasi.

Definisi turunan (limit):

$$f'(x) = \lim_{h\to 0} \frac{f(x+h) - f(x)}{h}$$

Berikut adalah beberapa sifat turunan fungsi yang dapat diperhatikan sehingga memudahkan dalam melakukan operasi turunan fungsi antara lain:

1. Aturan Penjumlahan

$$(f+g)'(x) = f'(x) + g'(x)]$$

2. Aturan Pengurangan

$$(f-g)'(x) = f'(x) - g'(x)]$$

3. Aturan Perkalian

$$(f \cdot g)'(x) = f'(x) \cdot g(x) + f(x) \cdot g'(x)]$$

4. Aturan Pembagian

$$(\frac{f}{g})'(x) = \frac{f'(x) \cdot g(x) + f(x) \cdot g'(x)}{(g(x))^2}$$

(dengan syarat g(x) tidak sama dengan 0)

5. Aturan Rantai
   Jika g(x) adalah fungsi dalam fungsi f(g(x))

$$(f(g(x)))' = f'(g(x)) \cdot g'(x)]$$

6. Turunan dari konstanta

$$\frac{d}{dx}(c) = 0, c = konstanta]1.AturanPenjumlahan$$

```
Contoh:

Terdapat f(x)=2x^2 dan g(x)=3x^2 tentukan nilai (f+g)'(x)
```

>$showev(’limit((2*(x+h)$^{2-2*x}$2)/h,h,0)+’limit((3*(x+h)$^{2-3*x}$2)/h,h,0))

$$\lim_{h\to 0}{\frac{-2\,x^2+2\,\left(h+x\right)^2}{h}}+\lim_{h\to 0}{\frac{-3\,x^2+3\,\left(h+x\right)^2}{h}}=10\,x$$

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

```
                                 2
                              2 x
```

>function g(x) &= 3*x^2

```
                                 2
                              3 x
```

>function h(x) &= 10*x

```
                              10 x
```

>plot2d([“f(x)”,“g(x)”,“h(x)”]):

2. Aturan Pengurangan
   Contoh:
   Terdapat f(x)=3x^2+3 dan g(x)=2x^2+2 tentukan nilai dari (f-g)’(x)

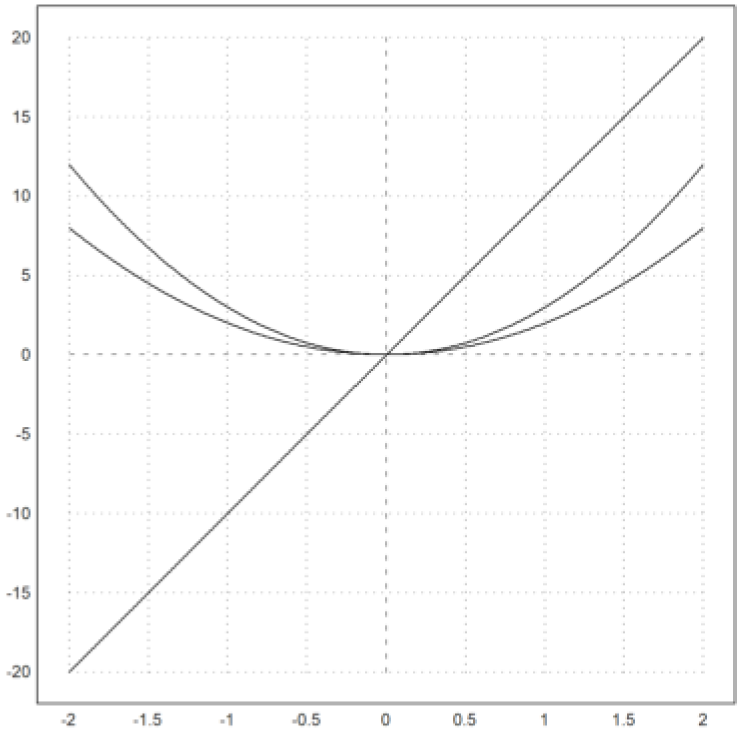
>$showev(’limit((3*(x+h)$^{2+3-(3*x}$2+3))/h,h,0)-’limit((2*(x+h)$^{2+2-(2*x}$2+2))/h,h,0))

$$-\lim_{h\to 0}{\frac{-2\,x^2+2\,\left(h+x\right)^2}{h}}+\lim_{h\to 0}{\frac{-3\,x^2+3\,\left(h+x\right)^2}{h}}=2\,x$$

>function f(x)&= 3*x^2+3

```
                                   2
                            3 + 3 x
```

>function g(x) &= 2*x^2+2



Gambar 344: images/KALKULUS_Diva%20Nagita(23030630024)-104.png

$$2 + 2\,x^2$$

>function h(x) &= 2*x

$$2\,x$$

>plot2d([“f(x)”,“g(x)”,“h(x)”]):

3. Aturan Perkalian
   Contoh:
   Terdapat u(x)=(x$^{3-2).(x}$2-3) tentukan nilai u’(x)
   Kita bisa menentukan terlebih dahulu f(x) dan g(x) dimana
   f(x)=xˆ3-2 dan g(x)=xˆ2-3 kemudian bisa kita terapkan aturan
   perkalian

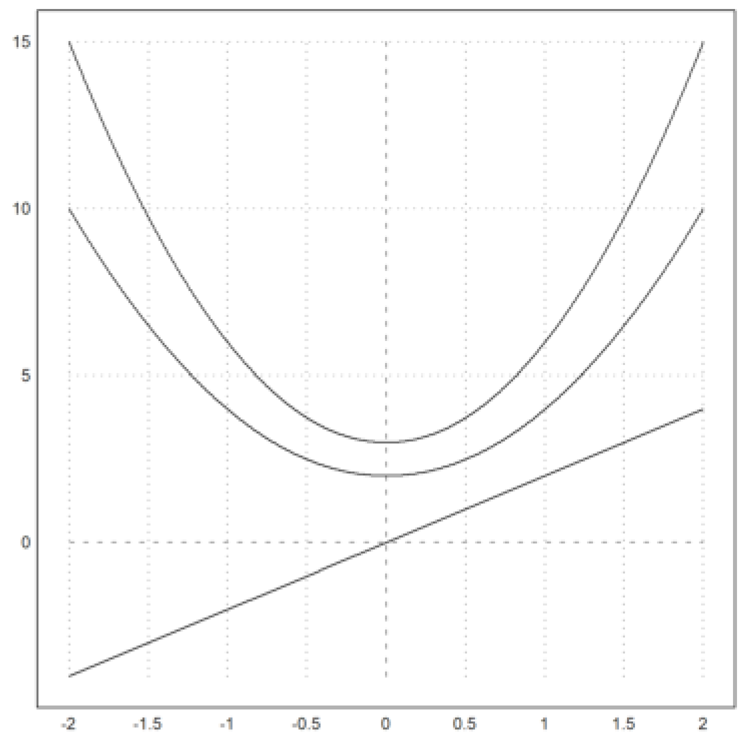
>$showev(((’limit(((x+h)$^{3-2-(x}$3-2))/h,h,0))*(x$^{2-3))+((x}$3-2)*’limit(((x+h)$^{2-3-(x}$2-3))/h,h,0)))

$$\left(-2+x^3\right)\,\left(\lim_{h\rightarrow 0}{\frac{-x^2+\left(h+x\right)^2}{h}}\right)+\left(-3+x^2\right)\,\left(\lim_{h\rightarrow 0}{\frac{-x^3+\left(h+x\right)^3}{h}}\right)=3\,x^2\,\left(-3+x^2\right)+2\,x\,\left(-2+x^3\right)$$

>$expand(((2*x)*(x$^{3-2))+((3*x}$2)*(xˆ2-3)))

$$-4\,x-9\,x^2+5\,x^4$$

>function u(x) &= (x$^{3-2)*(x}$2-3)



Gambar 345: images/KALKULUS_Diva%20Nagita(23030630024)-106.png

```
              2             3
(- 3 + x ) (- 2 + x )
```

>function h(x) &= 2*x*(x$^{3-2)+3*x}$2*(x^2-3)

```
  2         2                    3
3 x  (- 3 + x ) + 2 x (- 2 + x )
```

>plot2d(["u(x)","h(x)"]):

4. Aturan Pembagian
   Contoh:
   Terdapat u(x)=2x^3/x-1 tentukan nilai u'(x)
   Kita bisa menentukan terlebih dahulu f(x) dan g(x) dimana
   f(x)=2x^3 dan g(x)=x-1 kemudian bisa kita terapkan aturan pembagian

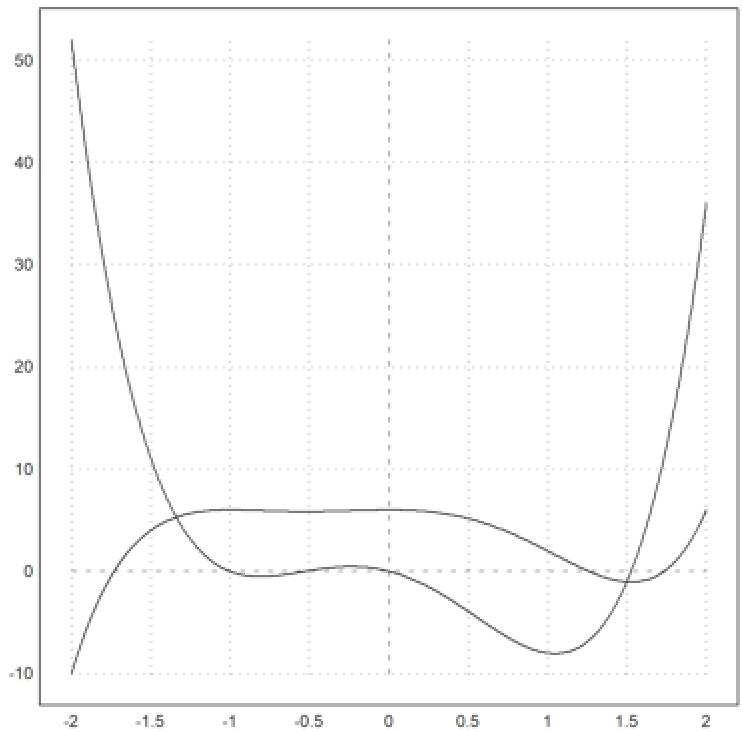
>$showev(((('limit((2*(x+h)$^{3-2*x}$3)/h,h,0))*(x-1))-((2*x$^{3)*'limit(((x+h)-1-(x-1))/h,h,0)))/(x-1)}$2)

$$\frac{-2\,x^3+(-1+x)\,\left(\lim_{h\to 0}\frac{-2\,x^3+2\,(h+x)^3}{h}\right)}{(-1+x)^2}=\frac{6\,(-1+x)\,x^2-2\,x^3}{(-1+x)^2}$$

>$expand((6*(x-1)*x^{2-2*x}3)/(x-1)^2)

$$-\frac{6\,x^2}{1-2\,x+x^2}+\frac{4\,x^3}{1-2\,x+x^2}$$

>function u(x)&= 2*x^3/(x-1)



Gambar 346: images/KALKULUS_Diva%20Nagita(23030630024)-109.png

```
      3
   2 x
  -------
  - 1 + x
```

>function h(x) &= (6*(x-1)*x^{2-2*x}3)/(x-1)^2

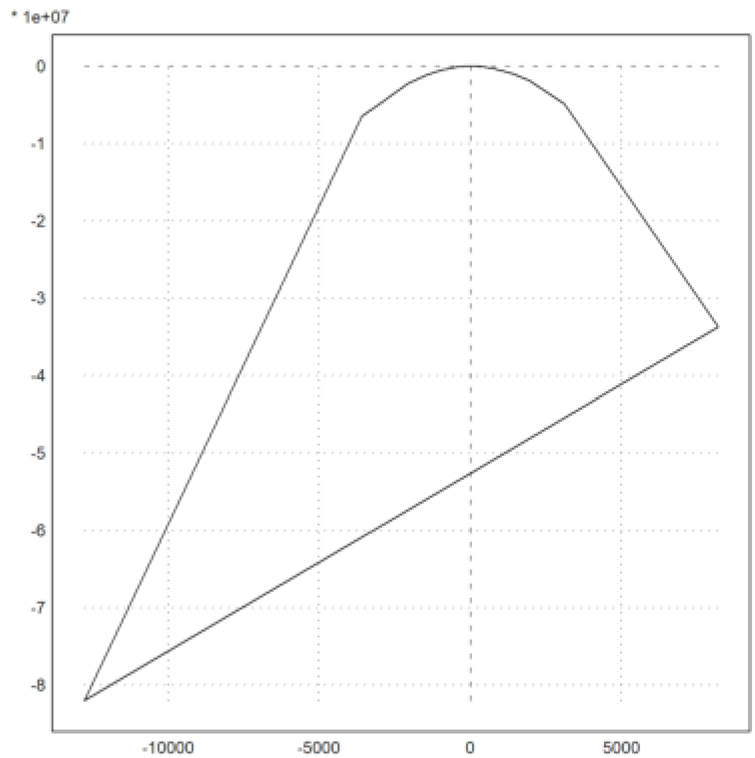
```
                     2       3
        6 (- 1 + x) x  - 2 x
        ---------------------
                       2
              (- 1 + x)
```

>plot2d("u(x)","h(x)",5.0):

5. Aturan rantai
   Contoh:
   Terdapat f(x)=x^2 dan g(x)=2x tentukan nilai dari (f(g(x)))'
   Kita bisa menentukan terlebih dahulu f(g(x)) yaitu
   f(g(x))=(2x)^2
   f(g(x))=4x^2
   Sehingga dapat kita gunakan aturan rantai sebagai berikut

>$showev(('limit((4*(x+h)$^{2-4*x}$2)/h,h,0))*('limit((2*(x+h)-2*x)/h,h,0)))

$$\left(\lim_{h\rightarrow 0}{\frac{-2\,x+2\,\left(h+x\right)}{h}}\right)\,\left(\lim_{h\rightarrow 0}{\frac{-4\,x^2+4\,\left(h+x\right)^2}{h}}\right)=16\,x$$



Gambar 347: images/KALKULUS_Diva%20Nagita(23030630024)-112.png

6. Turunan dari Konstanta
Contoh:
Cari nilai turunan dari 1000
>$showev('limit((1000-1000)/h,h,0))

$$0 = 0$$

Dalam mengetahui cara kerja limit untuk mencari turunan akan diberikan beberapa contoh antara lain:
Pada contoh pertama akan dicari turunan

$$x^2] > \$showev('limit(((x+h)^{2\text{-x}}2)/h,h,0))$$

$$\lim_{h\to 0}\frac{-x^2+(h+x)^2}{h} = 2\,x$$

pertama-tama bisa kita uraikan operasi pada pembilang

$$(x+h)^2 - x^2]$$

menguraikan

$$(x+h)^2]$$

menjadi

$$x^2 + 2hx + h^2]$$

kemudian disederhanakan dengan menghilangkan xˆ2

$$x^2 + 2hx + h^2 - x^2$$

menjadi

$$2hx + h^2] > p\& = expand((x+h)^{2\text{-x}}2)|simplify; \$p$$

$$h^2 + 2\,h\,x$$

Kemudian bentuk pecahan

$$\frac{2hx + h^2}{h}$$

dapat disederhanakan menjadi

$$2x + h$$

> q &=ratsimp(p/h); $q

$$h + 2\,x$$

Setelah itu bisa dimasukkan nilai limit sebagai turunan dimana h mendekati 0 sehingga didapat turunan dari

$$x^2]$$

adalah

$$2x] > \$limit(q, h, 0)$$

$$2\,x$$

Pada contoh kedua akan menentukan turunan dari

$$x^n] > \$showev('limit(((x+h)^{\text{n-x}}n)/h, h, 0))$$

$$\lim_{h\to 0}\frac{-x^n + (h+x)^n}{h} = n\,x^{-1+n}$$

Pada contoh ke 3 menentukan turunan dari sin(x)
>$showev('limit((sin(x+h)-sin(x))/h,h,0))

$$\lim_{h\to 0}\frac{-\sin x + \sin (h+x)}{h} = \cos x$$

Pada contoh berikutnya akan dicari turunan dari e^x
namun maxima bermasalah dengan limit:

$$\lim_{h\to 0}\frac{e^{x+h} - e^x}{h}.$$

Oleh karena itu perlu mengubah bentuk fungsi terlebih dahulu agar bisa mendapat hasil yang benar
>$showev('limit((E^h-1)/h,h,0))

$$\lim_{h\to 0}\frac{-1 + e^h}{h} = 1$$

dengan memfaktorkan pembilang menjadi:
>$showev('factor(E$^{(x+h)\text{-}E}$x))

$$factor\left(-e^x+e^{h+x}\right)=\left(-1+e^h\right)\,e^x$$

kemudian bisa kita masukkan hasil faktor tersebut kedalam limit, pada contoh, e^x merupakan konstanta terhadap limit sehingga bisa dikeluarkan

>$showev('limit(factor((E^(x+h)-E^x)/h),h,0)) // turunan f(x)=e^x

$$\left(\lim_{h\to 0}\frac{-1+e^h}{h}\right)\,e^x=e^x$$

berikut adalah beberapa contoh penggunaan limit pada turunan fungsi

1. turunan dari 1/x

>$showev('limit((1/(x+h)-1/x)/h,h,0))

$$\lim_{h\to 0}\frac{-\frac{1}{x}+\frac{1}{h+x}}{h}=-\frac{1}{x^2}$$

2. Turunan dari log(x)

>$showev('limit((log(x+h)-log(x))/h,h,0))

$$\lim_{h\to 0}\frac{-\log x+\log\left(h+x\right)}{h}=\frac{1}{x}$$

3. Turunan dari tan(x)

>$showev('limit((tan(x+h)-tan(x))/h,h,0))

$$\lim_{h\to 0}\frac{-\tan x+\tan\left(h+x\right)}{h}=\frac{1}{\cos^2 x}$$

4. Turunan dari arcsin(x)

>$showev('limit((asin(x+h)-asin(x))/h,h,0))

$$\lim_{h\to 0}\frac{-\arcsin x+\arcsin\left(h+x\right)}{h}=\frac{1}{\sqrt{1-x^2}}$$

Turunan Fungsi

Definisi turunan:

$$f'(x)=\lim_{h\to 0}\frac{f(x+h)-f(x)}{h}$$

Berikut adalah contoh-contoh menentukan turunan fungsi dengan menggunakan definisi turunan (limit).

>$showev('limit(((x+h)^2-x^2)/h,h,0)) // turunan x^2

$$\lim_{h\to 0}\frac{-x^2+\left(h+x\right)^2}{h}=2\,x$$

>p &= expand((x+h)^2-x^2) | simplify; $p //pembilang dijabarkan dan disederhanakan

$$h^2+2\,h\,x$$

>q &=ratsimp(p/h); $q // ekspresi yang akan dihitung limitnya disederhanakan

$$h+2\,x$$

>$limit(q,h,0) // nilai limit sebagai turunan

$$2\,x$$

>$showev('limit(((x+h)^n-x^n)/h,h,0)) // turunan x^n

$$\lim_{h\rightarrow 0}{\frac{-x^{n}+\left(h+x\right)^{n}}{h}}=n\,x^{-1+n}$$

>$showev('limit((sin(x+h)-sin(x))/h,h,0)) // turunan sin(x)

$$\lim_{h\rightarrow 0}{\frac{-\sin x+\sin \left(h+x\right)}{h}}=\cos x$$

>$showev('limit((log(x+h)-log(x))/h,h,0)) // turunan log(x)

$$\lim_{h\rightarrow 0}{\frac{-\log x+\log \left(h+x\right)}{h}}=\frac{1}{x}$$

>$showev('limit((1/(x+h)-1/x)/h,h,0)) // turunan 1/x

$$\lim_{h\rightarrow 0}{\frac{-\frac{1}{x}+\frac{1}{h+x}}{h}}=-\frac{1}{x^2}$$

>$showev('limit((E$^{(x+h)-E}$x)/h,h,0)) // turunan f(x)=eˆx

```
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Maxima is asking
Acceptable answers are: yes, y, Y, no, n, N, unknown, uk
Is x an integer?

Use assume!
Error in:
 $showev('limit((E^(x+h)-E^x)/h,h,0)) // turunan f(x)=e^x ...
                                     ^
```

Maxima bermasalah dengan limit:

$$\lim_{h\to 0}\frac{e^{x+h}-e^x}{h}.$$

Oleh karena itu diperlukan trik khusus agar hasilnya benar.

>$showev('limit((Eˆh-1)/h,h,0))

$$\lim_{h\rightarrow 0}{\frac{-1+e^{h}}{h}}=1$$

>$showev('factor(E$^{(x+h)-E}$x))

$$\mathit{factor}\left(-e^{x}+e^{h+x}\right)=\left(-1+e^{h}\right)\,e^{x}$$

>$showev('limit(factor((E$^{(x+h)-E}$x)/h),h,0)) // turunan f(x)=eˆx

$$\left(\lim_{h\rightarrow 0}{\frac{-1+e^{h}}{h}}\right)\,e^{x}=e^{x}$$

>function f(x) &= xˆx

```
                                  x
                                 x
```

>$showev('limit(f(x),x,0))

$$\lim_{x\to 0}x^x=1$$

>$showev('limit((f(x+h)-f(x))/h,h,0)) // turunan f(x)=x^x

$$\lim_{h\to 0}\frac{-x^x+(h+x)^{h+x}}{h}=\mathit{infinity}$$

Di sini Maxima juga bermasalah terkait limit:

$$\lim_{h\to 0}\frac{(x+h)^{x+h}-x^x}{h}.$$

Dalam hal ini diperlukan asumsi nilai x.

>&assume(x>0); $showev('limit((f(x+h)-f(x))/h,h,0)) // turunan f(x)=x^x

$$\lim_{h\to 0}\frac{-x^x+(h+x)^{h+x}}{h}=x^x\,(1+\log x)$$

>&forget(x>0) // jangan lupa, lupakan asumsi untuk kembali ke semula

```
[x &gt; 0]
```

>&forget(x<0)

```
[x &lt; 0]
```

>&facts()

```
[t &gt; 0, r &gt; 0]
```

>$showev('limit((asin(x+h)-asin(x))/h,h,0)) // turunan arcsin(x)

$$\lim_{h\to 0}\frac{-\arcsin x+\arcsin (h+x)}{h}=\frac{1}{\sqrt{1-x^2}}$$

>$showev('limit((tan(x+h)-tan(x))/h,h,0)) // turunan tan(x)

$$\lim_{h\to 0}\frac{-\tan x+\tan (h+x)}{h}=\frac{1}{\cos^2 x}$$

>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)

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

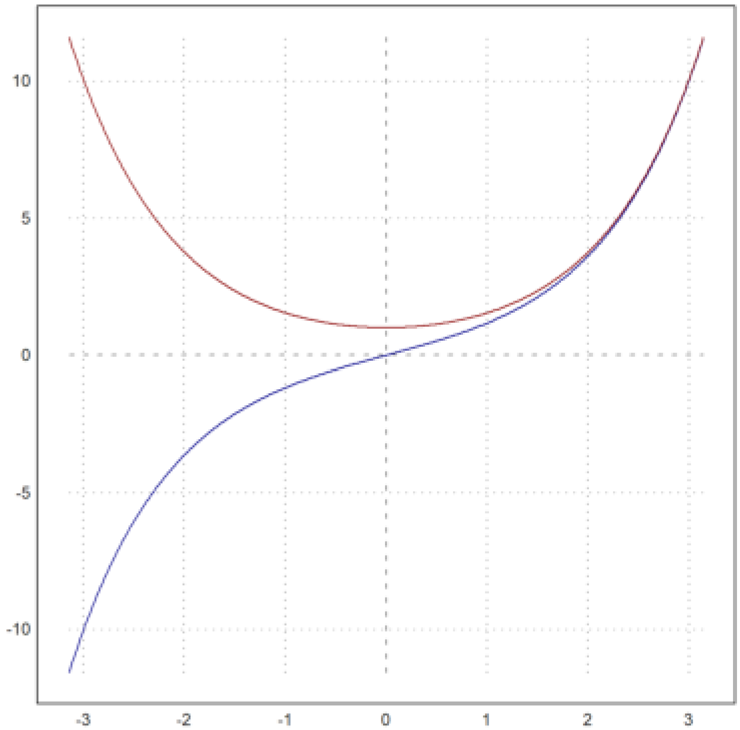
Hasilnya adalah cosh(x), karena

$$\frac{e^x+e^{-x}}{2}=\cosh(x).$$

>plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]):

>function f(x) &= sin(3*x$^{5+7)}$2

```
   2          5
sin (7 + 3 x )
```



Gambar 348: images/KALKULUS_Diva%20Nagita(23030630024)-161.png

>diff(f,3), diffc(f,3)

```
1198.32948904
1198.72863721
```

>$showev('diff(f(x),x))

$$\frac{d}{d\,x}\,\sin^2\left(7+3\,x^5\right)=30\,x^4\,\cos\left(7+3\,x^5\right)\,\sin\left(7+3\,x^5\right)$$

>$% with x=3

$$\%at\left(\frac{d}{d\,x}\,\sin^2\left(7+3\,x^5\right),x=3\right)=2430\,\cos 736\,\sin 736$$

>$float(%)

$$\%at\left(\frac{d^{1.0}}{d\,x^{1.0}}\,\sin^2\left(7.0+3.0\,x^5\right),x=3.0\right)=1198.728637211748$$

>plot2d(f,0,3.1):
>function f(x) &=5*cos(2*x)-2*x*sin(2*x) // mendifinisikan fungsi f

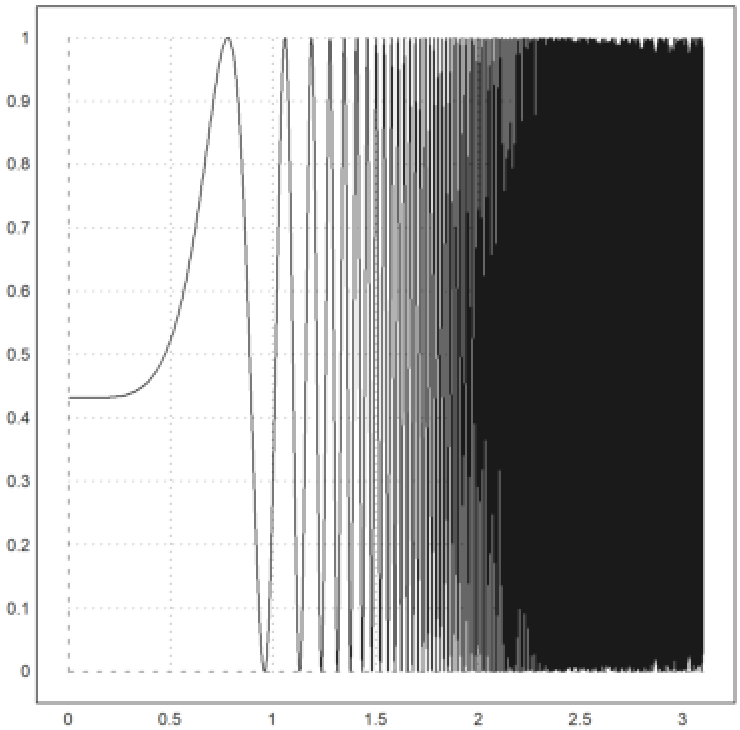
```
5 cos(2 x) - 2 x sin(2 x)
```

>function df(x) &=diff(f(x),x) // fd(x) = f'(x)

```
- 4 x cos(2 x) - 12 sin(2 x)
```

>$'f(1)=f(1), $float(f(1)), $'f(2)=f(2), $float(f(2)) // nilai f(1) dan f(2)

$$-0.2410081230863468$$



Gambar 349: images/KALKULUS_Diva%20Nagita(23030630024)-165.png

$$f(2) = 5\ \cos 4 - 4\ \sin 4$$

Gambar 350: images/KALKULUS_Diva%20Nagita(23030630024)-168.png

$$-0.2410081230863468$$

Gambar 351: images/KALKULUS_Diva%20Nagita(23030630024)-169.png

-3.899329036387075

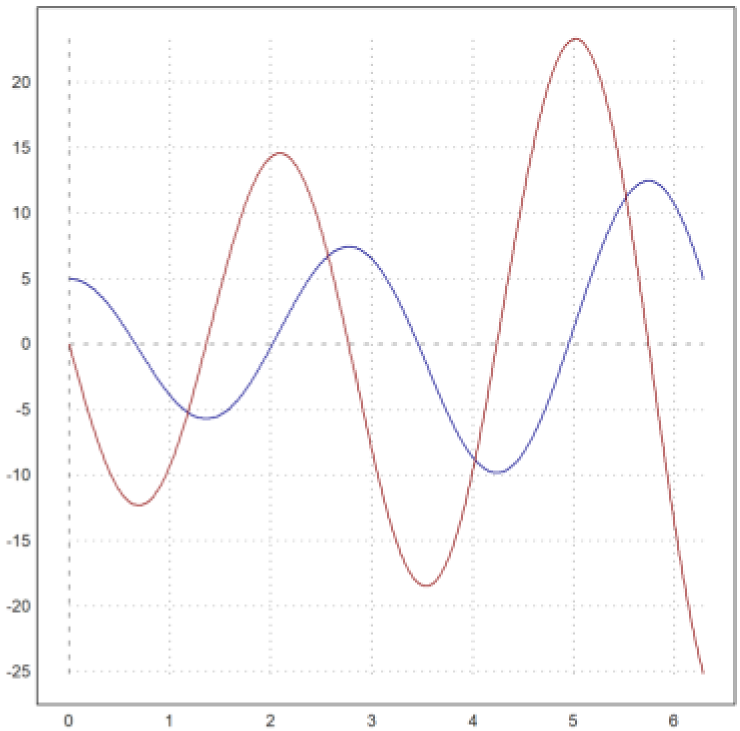
>xp=solve("df(x)",1,2,0) // solusi f'(x)=0 pada interval [1, 2]

```
1.35822987384
```

>df(xp), f(xp) // cek bahwa f'(xp)=0 dan nilai ekstrim di titik tersebut

```
0
-5.67530133759
```

>plot2d(["f(x)","df(x)"],0,2*pi,color=[blue,red]): //grafik fungsi dan turunannya



Gambar 352: images/KALKULUS_Diva%20Nagita(23030630024)-170.png

Visualisasi Grafik dan Penggunaan Turunan Fungsi

Penggunaan Turunan fungsi antara lain
Menentukan Gradien Garis Singgung
Dimana jika terdapat fungsi y=f(x), maka gradien pada titik tertentu dapat dinyatakan sebagai m=f'(x)
sebagai contoh, kita akan cari garis singgung dari suatu grafik fungsi

$$x^2 - 2x + 1] > function f(x) \& = x^2 - 2 * x + 1$$

```
              2
 1 - 2 x + x
```

>function df(x) &=diff(f(x),x) // fd(x) = f'(x)

```
 - 2 + 2 x
```

Setelah mengetahui turunan fungsi tersebut, kita coba memasukkan nilai x=2 maka didapat

$$m = 2(2) - 2] \; m = 2] kemudiankitamasukkannilaixpadafungsiawaluntukmendapattitiky$$

yaitu

$$f(2) = 2^2 - 2.2 + 1] \; f(2) = 1] didapattitiksinggungpadagrafikadalah(2, 1)dengangradiengarism = 2$$

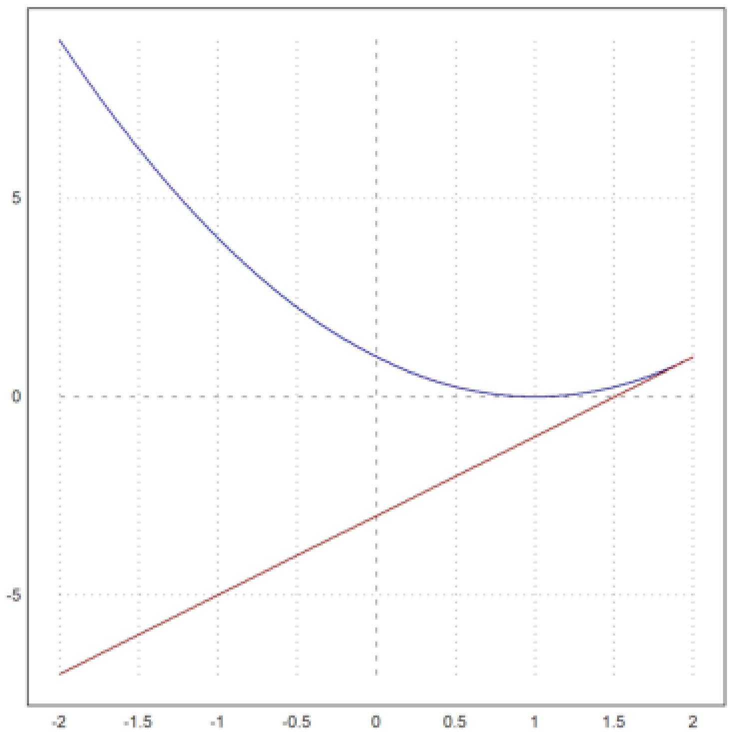
yang kemudian dapat kita cari persamaan garisnya:

$$y - y_1 = m(x - x_1)] \; y - 1 = 2(x - 2)] \; y - 1 = 2x - 4] \; y = 2x - 3] yangkemudiandapatkitavisualisasikangrafiknya$$

>function g(x) &= 2*x-3

```
- 3 + 2 x
```

>plot2d(["f(x)","g(x)"],color=[blue,red]):



Gambar 353: images/KALKULUS_Diva%20Nagita(23030630024)-180.png

---

Pada turunan pertama fungsi dapat kita gunakan pula untuk menentukan titik kritis suatu fungsi dimana

$$f'(x) = a]$$

a adalah absis

Pada turunan kedua fungsi juga dapat kita gunakan untuk menentukan titik maksimum lokal atau titik minimum lokal suatu fungsi dimana c adalah titik kritis

$$f"(c) > 0]$$

maka c titik minimum lokal

$$f"(c) < 0]$$

maka c titik maksimum lokal

Berikut adalah contoh lain yaitu fungsi

$$f(x) = 5cos(2x) - 2sin(2x)] > functionf(x)\& = 5 * cos(2 * x) - 2 * x * sin(2 * x)$$

```
5 cos(2 x) - 2 x sin(2 x)
```

>function df(x) &=diff(f(x),x) // fd(x) = f'(x)

```
- 4 x cos(2 x) - 12 sin(2 x)
```

nilai f(1) dan f(2)
>$'f(1)=f(1), $float(f(1)), $'f(2)=f(2), $float(f(2))

$$-0.2410081230863468$$

-3.899329036387075

$$f(2) = 5\cos 4 - 4\sin 4$$

Gambar 354: images/KALKULUS_Diva%20Nagita(23030630024)-187.png

$$-0.2410081230863468$$

Gambar 355: images/KALKULUS_Diva%20Nagita(23030630024)-188.png

solusi f'(x)=0 pada interval [1, 2]
>xp=solve("df(x)",1,2,0)

```
1.35822987384
```

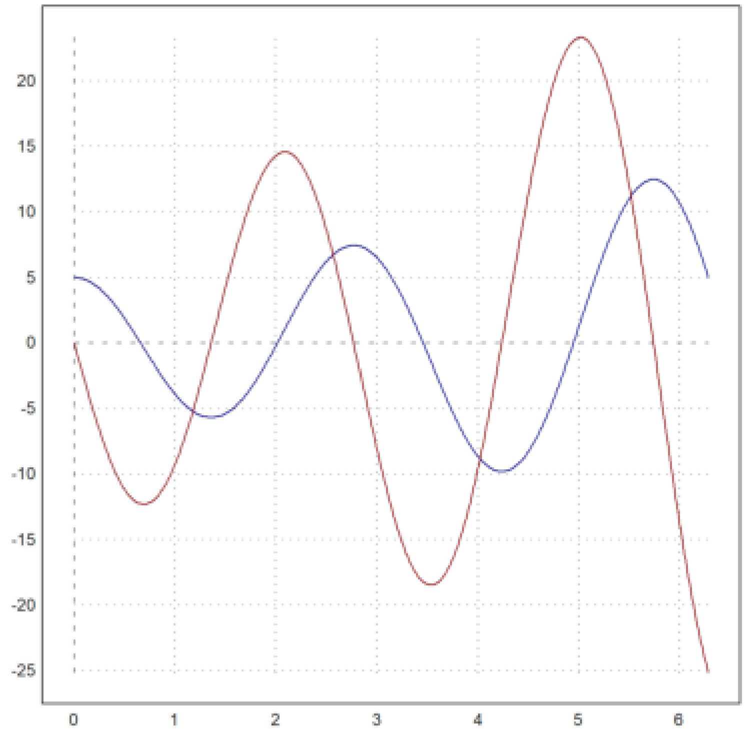
cek bahwa f'(xp)=0 dan nilai ekstrim di titik tersebut
>df(xp), f(xp)

```
0
-5.67530133759
```

grafik fungsi dan turunannya
>plot2d(["f(x)","df(x)"],0,2*pi,color=[blue,red]):
Contoh lain yaitu pada fungsi

$$f(x) = sinh(x)] > functionf(x)\& = sinh(x)$$

```
sinh(x)
```



Gambar 356: images/KALKULUS_Diva%20Nagita(23030630024)-189.png

>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x) // df(x) = f'(x)

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

Didapat hasilnya adalah cosh(x), karena

$$\frac{e^x+e^{-x}}{2}=\cosh(x).$$

>plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]):

Integral

Jumlah Riemann (Riemann Sum)

Konsep dasar dari integral adalah menghitung luas di bawah kurva suatu fungsi. Untuk mendekati luas ini secara numerik, kita dapat membagi wilayah di bawah kurva menjadi persegi panjang-persegi panjang kecil, dan kemudian menjumlahkan luasnya. Ini disebut jumlah Riemann, yang menggunakan notasi sigma

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

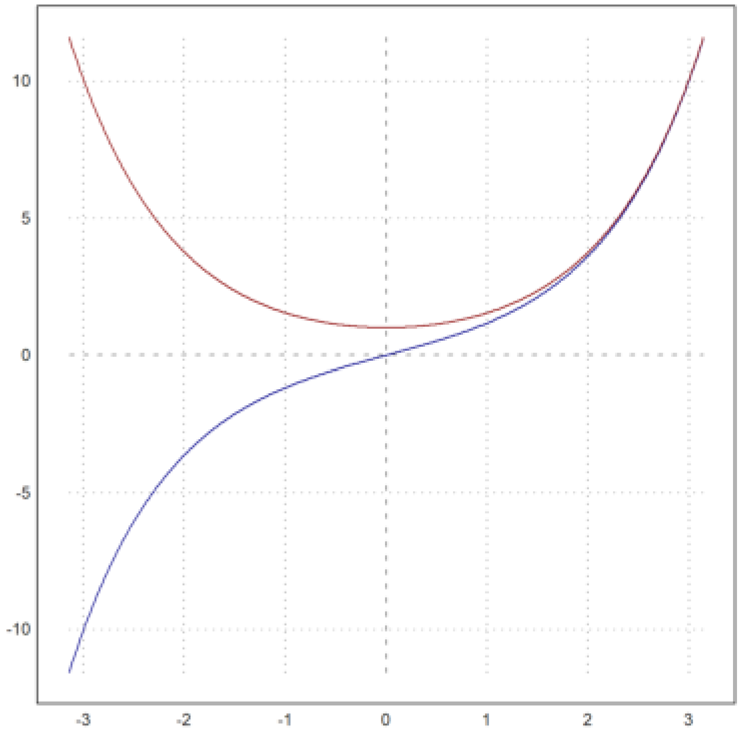
>x=0:0.1:pi-0.1; plot2d(x,f(x+0.1),>bar); plot2d("f(x)",0,pi,>add):

Misalkan kita ingin menghitung luas di bawah kurva fungsi

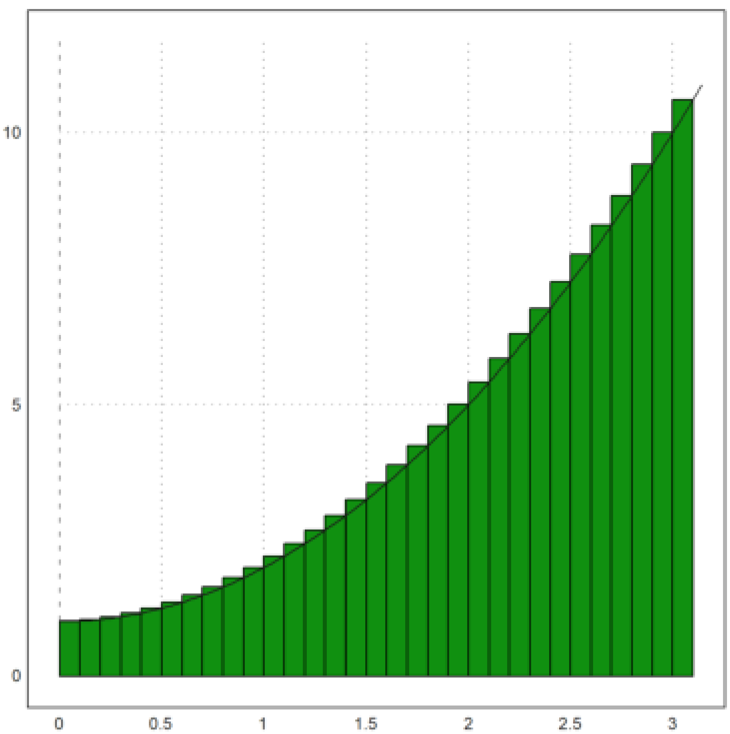
f(x) dari x=a hingga x=b. Untuk mendekati luas ini, kita membagi interval[a,b] menjadi n subinterval yang sama panjang, di mana panjang setiap subinterval adalah:

$$\delta x=\frac{b-a}{n}$$

Kemudian kita pilih titik xi dalam setiap subinterval sebagai titik representatif. Nilai f(xi) digunakan untuk menentukan tinggi persegi panjang, dan delta x sebagai lebarnya.



Gambar 357: images/KALKULUS_Diva%20Nagita(23030630024)-193.png



Gambar 358: images/KALKULUS_Diva%20Nagita(23030630024)-194.png

Jumlah Riemann didefinisikan sebagai:

image: Rumus_Reimann

Ketika kita meningkatkan jumlah subinterval (yakni n menuju tak hingga), maka pendekatan jumlah Riemann ini semakin mendekati nilai integral. Dengan kata lain, integral tentu dari fungsi

f(x) dari a hingga b didefinisikan sebagai limit dari jumlah Riemann:

image: Rumus_Hubungan_Integral

Syarat integral

Fungsi Kontinu

Secara matematis, fungsi f(x) dikatakan kontinu pada interval[a,b] jika memenuhi syarat-syarat berikut:

Definisi Kekontinuan fungsi(x) kontinu pada interval [a,b] jika: + Terdefinisi: Fungsi harus terdefinisi di setiap titik dalam interval tersebut. Artinya, untuk setiap nilai x dalam interval[a,b],f(x) harus memiliki nilai yang terdefinisi. + Limit Ada: Limit fungsi harus ada pada setiap titik dalam interval. + Untuk setiap titik dalam interval [a,b], limit x=>c f(x) harus ada. + Limit Sama dengan Nilai Fungsi: Nilai fungsi pada titik tersebut harus sama dengan limit fungsi ketika mendekati titik tersebut. Yaitu, lim=>c f(x)=f(c).

Diketahui sebuah fungsi

$$h(x) = 1/x$$

dari x=-2 hingga x=2

>function h(x):=1/x

>h(-2)

```
-0.5
```

>h(-1)

```
-1
```

>h(0)

```
Floating point error!
Try "trace errors" to inspect local variables after errors.
h:
    useglobal; return 1/x
Error in:
h(0) ...
    ^
```

Diketahui fungsi f(x)

$$f(x) = 2x^3 - 3x + 1$$

dari x=1 hingga x=2

>function f(x):=2*x^3-3*x+1

>f(1)

```
0
```

>f(2)

```
11
```

>$limit(2*x^3-3*x+1,x,1)

$$0$$

>$limit(2*x^3-3*x+1,x,2)

$$11$$

Integral

EMT dapat digunakan untuk menghitung integral, baik integral tak tentu maupun integral tentu. Untuk integral tak tentu (simbolik) sudah tentu EMT menggunakan Maxima, sedangkan untuk perhitungan integral tentu EMT sudah menyediakan beberapa fungsi yang mengimplementasikan algoritma kuadratur (perhitungan integral tentu menggunakan metode numerik).

Pada notebook ini akan ditunjukkan perhitungan integral tentu dengan menggunakan Teorema Dasar Kalkulus:

$$\int_a^b f(x)\,dx = F(b) - F(a), \quad \text{dengan } F'(x) = f(x).$$

Fungsi untuk menentukan integral adalah integrate. Fungsi ini dapat digunakan untuk menentukan, baik integral tentu maupun tak tentu (jika fungsinya memiliki antiderivatif). Untuk perhitungan integral tentu fungsi integrate menggunakan metode numerik (kecuali fungsinya tidak integrabel, kita tidak akan menggunakan metode ini).

Integral Tak Tentu

EMT dapat digunakan untuk menghitung integral, baik integral tak tentu maupun integral tentu. Untuk integral tak tentu (simbolik) sudah tentu EMT menggunakan Maxima.

Pada notebook ini akan ditunjukkan perhitungan integral tentu dengan menggunakan Teorema Dasar Kalkulus:

$$\int f(x)\,dx = F(x) + C$$

Fungsi untuk menentukan integral adalah integrate. Fungsi ini dapat digunakan untuk menentukan, baik integral tentu maupun tak tentu (jika fungsinya memiliki antiderivatif).

>function f(x):=(1/x^2+1)

>x=0:0.1:pi-0.1; plot2d(x,f(x+0.1),>bar); plot2d("f(x)",0,pi,>add):

>$showev('integrate(x^n,x))

```
Answering "Is n equal to -1?" with "no"
```

$$\int x^n\,dx = \frac{x^{1+n}}{1+n}$$

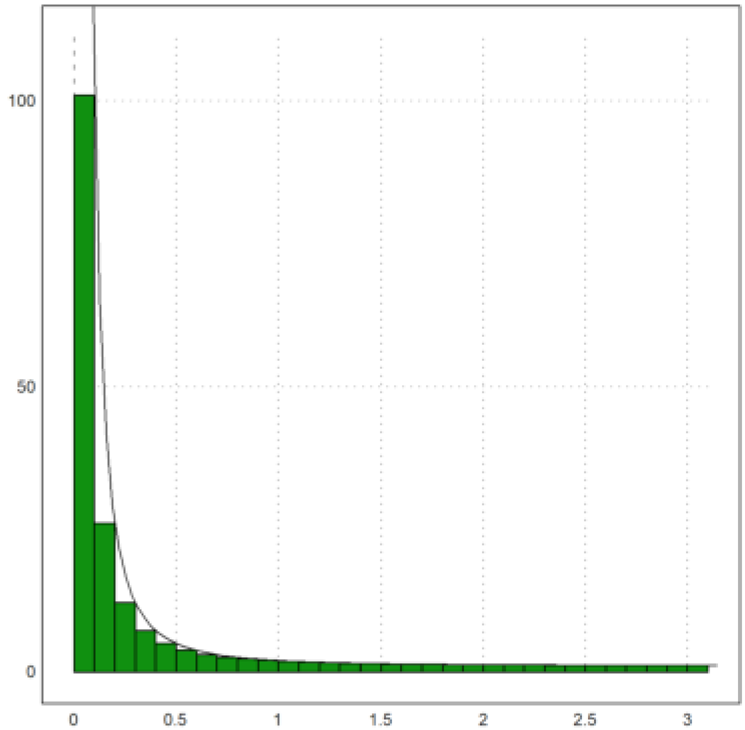
>$showev('integrate(1/(1+x^2),x))

$$\int \frac{1}{1+x^2}\,dx = \arctan x$$

>$showev('integrate(1/sqrt(1-x^2),x))

$$\int \frac{1}{\sqrt{1-x^2}}\,dx = \arcsin x$$

>$showev('integrate(sin(x),x,0,pi))



Gambar 359: images/KALKULUS_Diva%20Nagita(23030630024)-202.png

$$\int_{0}^{\pi}{\sin x\;dx}=2$$

>plot2d("sin(x)",0,2*pi):

>$showev('integrate((y^2),y))

$$\int {y^2\;dy}=\frac{y^3}{3}$$

>function f(x):=(x^{5+3*x}2+x+3)

>x=0:0.1:pi-0.1; plot2d(x,f(x+0.1),>bar); plot2d("f(x)",0,pi,>add):

>$showev('integrate(sin(x),x,a,b))

$$\int_{a}^{b}{\sin x\;dx}=\cos a-\cos b$$

>$showev('integrate(x^n,x,a,b))

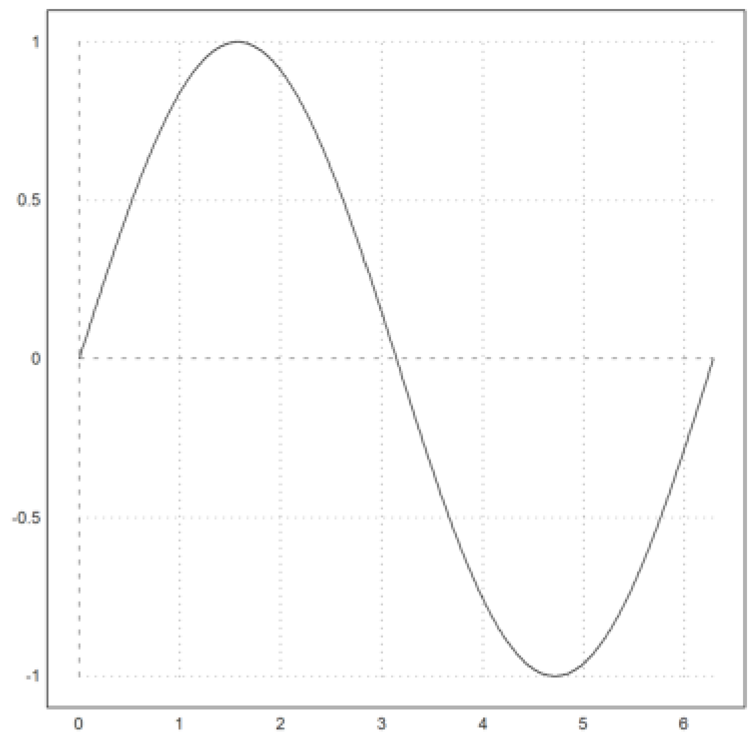
```
Answering "Is n positive, negative or zero?" with "positive"
```

$$\int_{a}^{b}{x^{n}\;dx}=-\frac{a^{1+n}}{1+n}+\frac{b^{1+n}}{1+n}$$

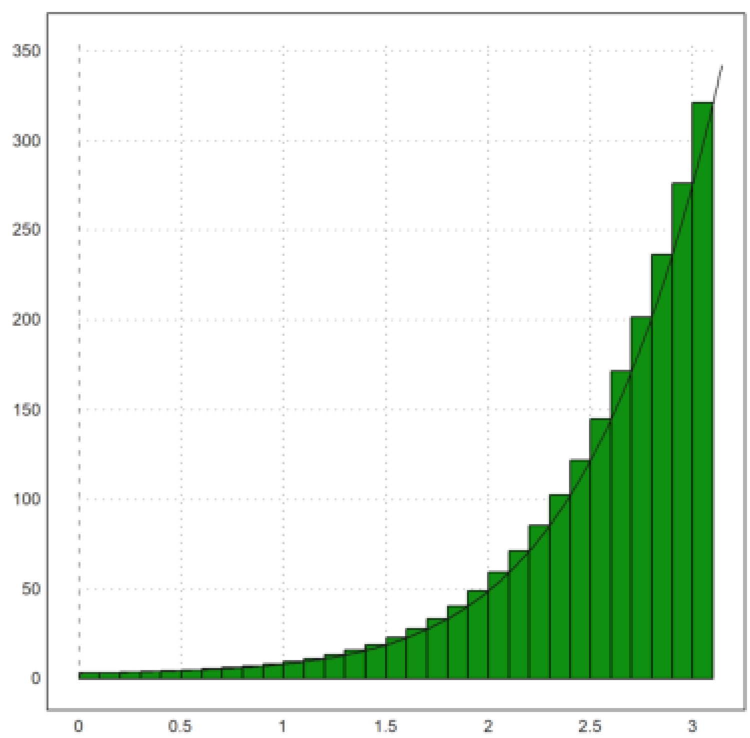
>$showev('integrate(x^2*sqrt(2*x+1),x,0,2))

$$\int_{0}^{2}{x^2\,\sqrt{1+2\,x}\;dx}=-\frac{2}{105}+\frac{2\,5^{\frac{5}{2}}}{21}$$

>$ratsimp(%)



Gambar 360: images/KALKULUS_Diva%20Nagita(23030630024)-207.png



Gambar 361: images/KALKULUS_Diva%20Nagita(23030630024)-209.png

$$\int_{0}^{2}{x^2\,\sqrt{1+2\,x}\;dx}=\frac{-2+2\,5^{\frac{7}{2}}}{105}$$

>$showev('integrate(x^5+3*x^2+x+3, x))

$$\int {3+x+3\,x^2+x^5}{\;dx}=3\,x+\frac{x^2}{2}+x^3+\frac{x^6}{6}$$

>$showev('integrate(3*x^3-2*x,x))

$$\int {-2\,x+3\,x^3}{\;dx}=-x^2+\frac{3\,x^4}{4}$$

>$showev('integrate((sin(sqrt(x)+a)*E^sqrt(x))/sqrt(x),x,0,pi^2))

$$\int_{0}^{\pi^2}{\frac{\sin \left(a+\sqrt{x}\right)\,e^{\sqrt{x}}}{\sqrt{x}}\;dx}=\left(1+e^{\pi}\right)\,\cos a+\left(-1-e^{\pi}\right)\,\sin a$$

>$factor(%)

$$\int_{0}^{\pi^2}{\frac{\sin \left(a+\sqrt{x}\right)\,e^{\sqrt{x}}}{\sqrt{x}}\;dx}=\left(-1-e^{\pi}\right)\,\left(-\cos a+\sin a\right)$$

>$showev('integrate(x^2*sqrt(2*x+1),x))

$$\int {x^2\,\sqrt{1+2\,x}}{\;dx}=\frac{\left(1+2\,x\right)^{\frac{3}{2}}}{12}-\frac{\left(1+2\,x\right)^{\frac{5}{2}}}{10}+\frac{\left(1+2\,x\right)^{\frac{7}{2}}}{28}$$

>$showev('integrate (cos(2*x),x))

$$\int {\cos \left(2\,x\right)}{\;dx}=\frac{\sin \left(2\,x\right)}{2}$$

>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.

Fungsi error, dinotasikan sebagai erf(x), didefinisikan sebagai integral tertentu berikut:

$$erf(x)=\int \frac{e^{-x^2}}{\sqrt{\pi}}\,dx.$$

Integral tentu

Integral tentu adalah integral matematika yang memiliki batasan atas dan bawah yang jelas, sehingga menghasilkan sebuah nilai. Batasan dari integral tentu adalah a sampai b atau batas atas sampai batas bawah.

Untuk perhitungan integral tentu fungsi integrate menggunakan metode numerik (kecuali fungsinya tidak integrabel, kita tidak akan menggunakan metode ini).

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).

$$\int_a^b (f(x))dx$$

$$F(b) - F(a)$$

$$\int_0^{\infty} e^{-x^2}\, dx$$

dapat dihampiri dengan jumlah luas persegi-persegi panjang di bawah kurva y=f(x) tersebut. Langkah-langkahnya adalah sebagai berikut.

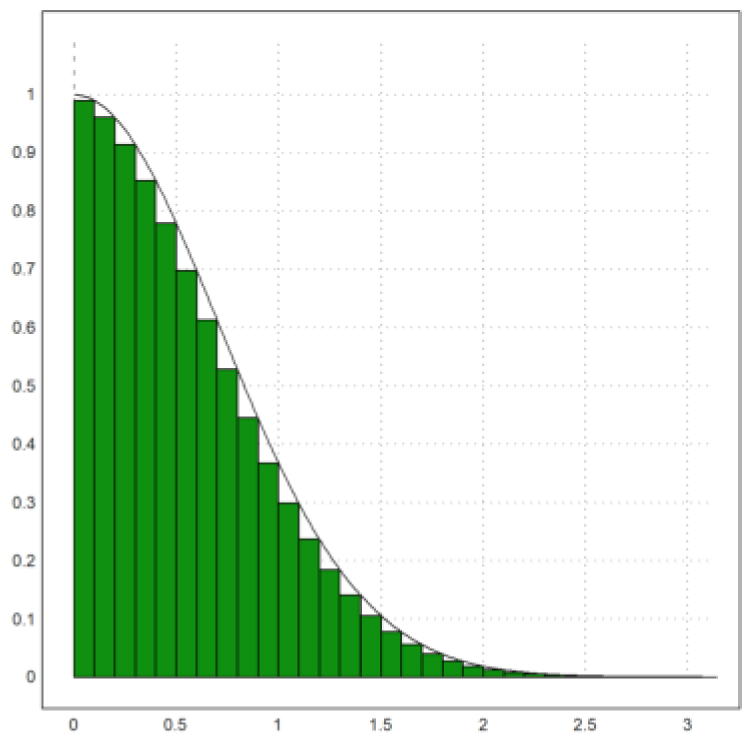
> function map f(x) &= E^(-x^2)

```
          2
       - x
     E
```

>$showev('integrate(f(x),x))

$$\int e^{-x^2}\, dx = \frac{\sqrt{\pi}\,\mathrm{erf}\left(x\right)}{2}$$

>x=0:0.1:pi-0.1; plot2d(x,f(x+0.1),>bar); plot2d("f(x)",0,pi,>add):



Gambar 362: images/KALKULUS_Diva%20Nagita(23030630024)-226.png

>t &= makelist(a,a,0,pi-0.1,0.1); // t sebagai list untuk menyimpan nilai-nilai x

>fx &= makelist(f(t[i]+0.1),i,1,length(t)); // simpan nilai-nilai f(x)

>$showev('integrate(f(x),x,0,inf))

$$\int_0^{\infty} e^{-x^2}\, dx = \frac{\sqrt{\pi}}{2}$$

* fx &=: Ini adalah operasi penugasan gabungan. Artinya, hasil dari perhitungan di sebelah kanan akan ditambahkan ke variabel fx. Jika fx belum didefinisikan sebelumnya, maka fx akan dibuat sebagai sebuah list.
* -makelist(...,i,1,length(t)): Fungsi ini digunakan untuk membuat * sebuah list.

Parameter-parameternya adalah:

- f(t[i]+0.1): Ini adalah ekspresi yang akan dihitung untuk setiap nilai i. Fungsi f akan dievaluasi pada nilai t[i] ditambah 0.1.
- i: Adalah indeks yang akan digunakan untuk mengakses elemen-elemen dalam list t.
- 1: Adalah nilai awal untuk indeks i.
- length(t): Adalah nilai akhir untuk indeks i, yaitu panjang dari list t.

>$showev('integrate(x*exp(-x),x,0,1))

$$\int_{0}^{1}{x\,e^ {- x }\;dx}=1-2\,e^ {- 1 }$$

tentukan integral dari

$$3x^2 + 4x + 2$$

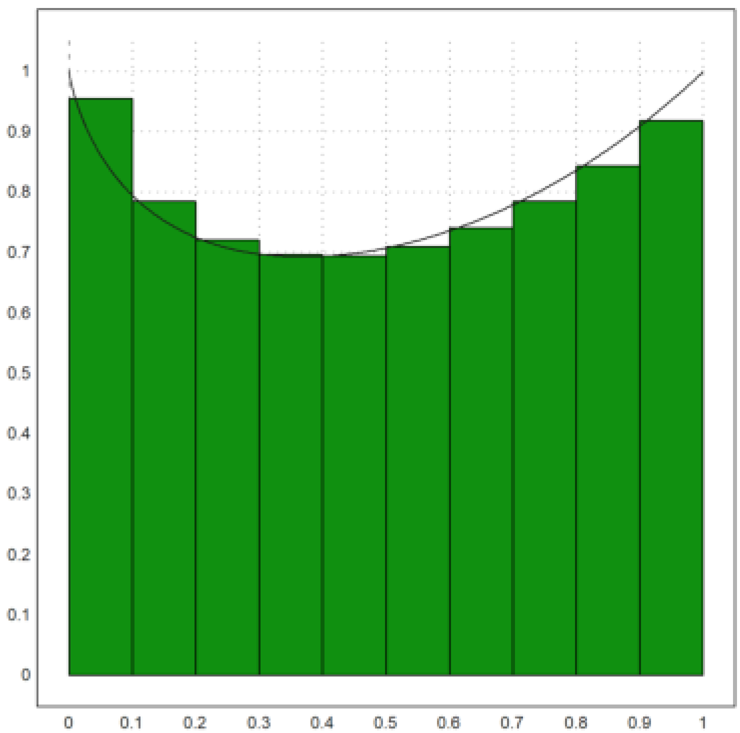
darix=1 sampai x=2
>function f(x) &= x^x

```
 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):



Gambar 363: images/KALKULUS_Diva%20Nagita(23030630024)-231.png

>function f(x) &= sin(3*x$^{5+7)}$2

```
                                2        5
                             sin (7 + 3 x )
```

>integrate(f,0,1)

```
0.542581176074
```

>&showev('integrate(f(x),x,0,1))

```
         1                            1              pi
        /                      gamma(-) sin(14) sin(--)
        [     2        5              5              10
        I  sin (7 + 3 x ) dx = ------------------------
        ]                                   1/5
        /                                10 6
         0
              4/5                       1
 - (- 60 + ((- 6    I gamma_incomplete(-, - 6 I)
                                        5
    4/5                    1
 + 6    I gamma_incomplete(-, 6 I)) cos(14)
                           5
     4/5                  1             4/5                 1
 + (6    gamma_incomplete(-, - 6 I) + 6    gamma_incomplete(-, 6 I))
                          5                                 5
             pi
 sin(14)) sin(--))/120
             10
```

>&float(%)

```
         1.0
        /
        [       2            5
        I    sin (7.0 + 3.0 x ) dx =
        ]
        /
         0.0
0.09820784258795788 - 0.008333333333333333
 (- 60.0 + 0.3090169943749474 (0.9906073556948704
 (4.192962712629476 gamma__incomplete(0.2, - 6.0 I)
 + 4.192962712629476 gamma__incomplete(0.2, 6.0 I))
 + 0.1367372182078336 (- 4.192962712629476 I
 gamma__incomplete(0.2, - 6.0 I) + 4.192962712629476 I
 gamma__incomplete(0.2, 6.0 I))))
```

>$showev('integrate(x*exp(-x),x,0,1)) // Integral tentu (eksak)

$$\int_{0}^{1}{x\,e^ {- x }\;dx}=1-2\,e^ {- 1 }$$

>function h(x):=3*x^2+4*x+2
>$showev('integrate (3*x^2+4*x+2,x,1,2))

$$\int_{1}^{2}{2+4\,x+3\,x^2\;dx}=15$$

>$showev('integrate (ln(x),x,1,2))

$$\int_{1}^{2}\log x\;dx=-1+2\,\log 2$$

>$showev('integrate (sin(x),x,0,pi))

$$\int_{0}^{\pi}\sin x\;dx=2$$

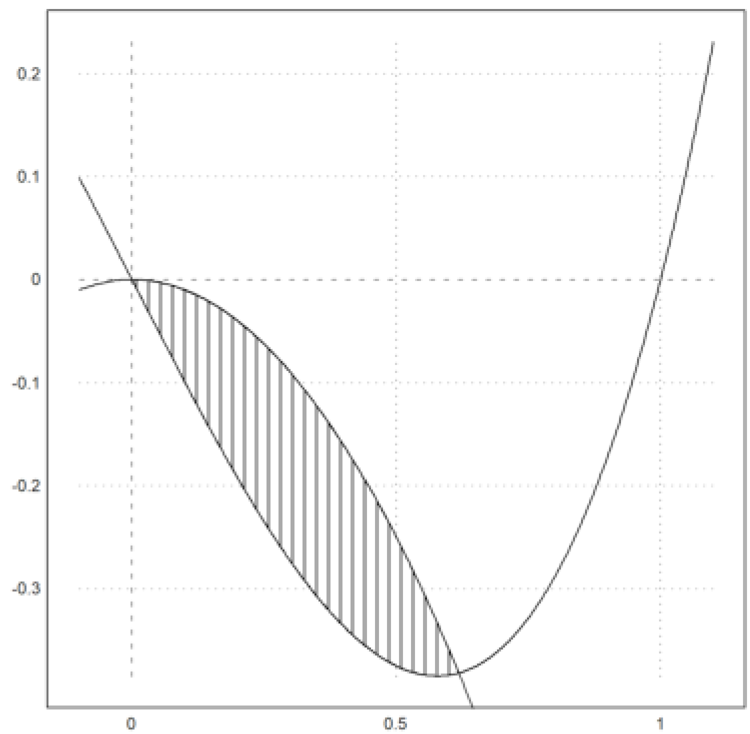
>$showev('integrate (cos(2*x),x))

$$\int\cos\left(2\,x\right)\;dx=\frac{\sin\left(2\,x\right)}{2}$$

>reset;

Aplikasi Integral Tentu

>plot2d("xˆ3-x",-0.1,1.1); plot2d("-xˆ2",>add); ...
> b=solve("x^{3-x+x}2",0.5); x=linspace(0,b,200); xi=flipx(x); ...
> plot2d(x|xi,x$^{3-x|-xi}$2,>filled,style="|",fillcolor=1,>add): // Plot daerah antara 2 kurva



Gambar 364: images/KALKULUS_Diva%20Nagita(23030630024)-237.png

>a=solve("x^{3-x+x}2",0), b=solve("x^{3-x+x}2",1) // absis titik-titik potong kedua kurva

```
0
0.61803398875
```

>integrate("(-x$^{2)-(x}$3-x)",a,b) // luas daerah yang diarsir

```
0.0758191713542
```

>a &= solve((-x$^{2)-(x}$3-x),x); $a // menentukan absis titik potong kedua kurva secara eksak

$$\left[x=\frac{-\sqrt{5}-1}{2},x=\frac{\sqrt{5}-1}{2},x=0\right]$$

>$showev('integrate(-x^{2-x}3+x,x,0,(sqrt(5)-1)/2)) // Nilai integral secara eksak

$$\int_{0}^{\frac{\sqrt{5}-1}{2}}{-x^3-x^2+x\;dx}=\frac{13-5^{\frac{3}{2}}}{24}$$

>$float(%)

$$\int_{0.0}^{0.6180339887498949}{-1.0\,x^3-1.0\,x^2+x\;dx}=0.07581917135421037$$

1. Panjang Kurva

Panjang kurva dari kurva

y=f(x) pada interval [a,b]dapat dihitung dengan menggunakan integral. Panjang kurva L dapat didefinisikan sebagai:

image: Rumus_Panjang_Kurva

Tentukan panjang kurva

$$y=x^2$$

dari x=0 hingga x=2

>&diff(x^2, x)

```
2 x
```

>$showev('integrate (sqrt(1+(2*x)^2), x, 0, 2))

$$\int_{0}^{2}{\sqrt{4\,x^2+1}\;dx}=\frac{\operatorname{asinh}4+4\,\sqrt{17}}{4}$$

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 dahulu

>function r(t) &= 1+sin(3*t)/2; $'r(t)=r(t)

$$r\left(t\right)=\frac{\sin\left(3\,t\right)}{2}+1$$

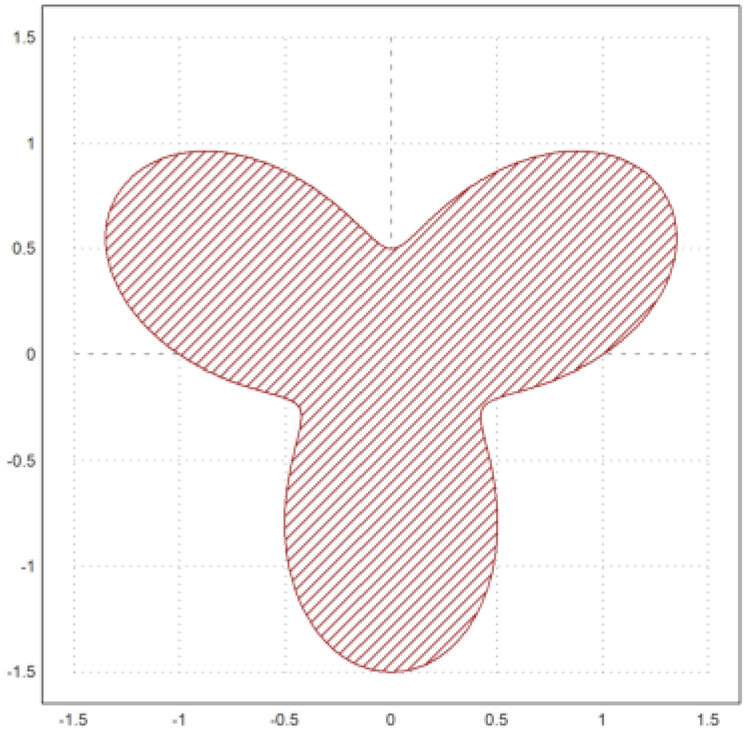
>function fx(t) &= r(t)*cos(t); $'fx(t)=fx(t)

$$\mathit{fx}\left(t\right)=\cos t\,\left(\frac{\sin\left(3\,t\right)}{2}+1\right)$$

>function fy(t) &= r(t)*sin(t); $'fy(t)=fy(t):

>function ds(t) &= trigreduce(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



Gambar 365: images/KALKULUS_Diva%20Nagita(23030630024)-245.png

$$\frac{\int_0^{2\,\pi} \sqrt{4\,\cos\left(6\,x\right)+4\,\sin\left(3\,x\right)+9}\,dx}{2}$$

>integrate("ds(x)",0,2*pi)

```
9.0749467823
```

Spiral Logaritmik

$$x=e^{ax}\cos x,\ y=e^{ax}\sin x.$$

>a=0.1; plot2d("exp(a*x)*cos(x)","exp(a*x)*sin(x)",r=2,xmin=0,xmax=2*pi):

Koordinat Kartesius

Perhitungan panjang kurva menggunakan koordinat Kartesius. Kita akan hitung panjang kurva dengan persamaan implisit:

$$x^3+y^3-3xy=0.$$

>z &= x^{3+y}3-3*x*y; $z

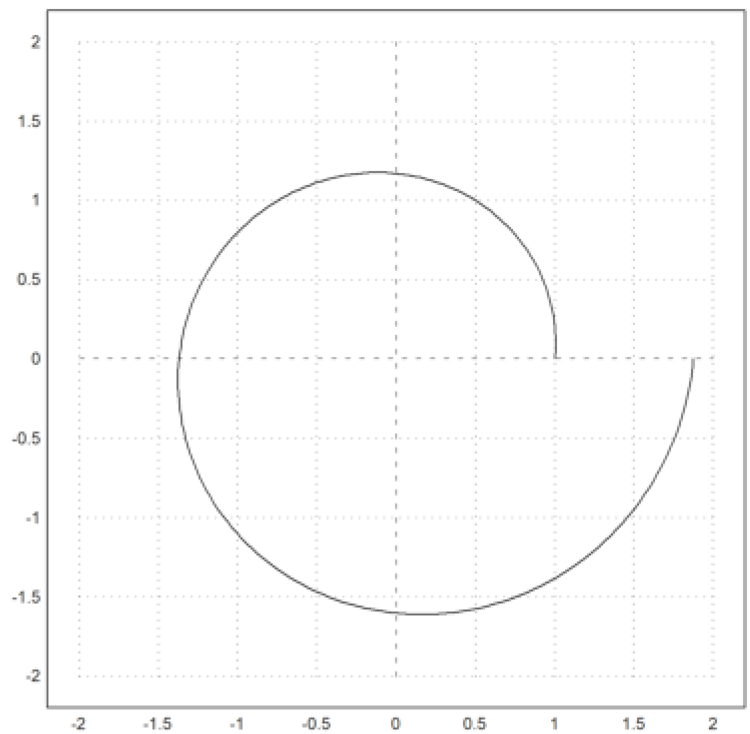
$$y^3-3\,x\,y+x^3$$

>plot2d(z,r=2,level=0,n=100):

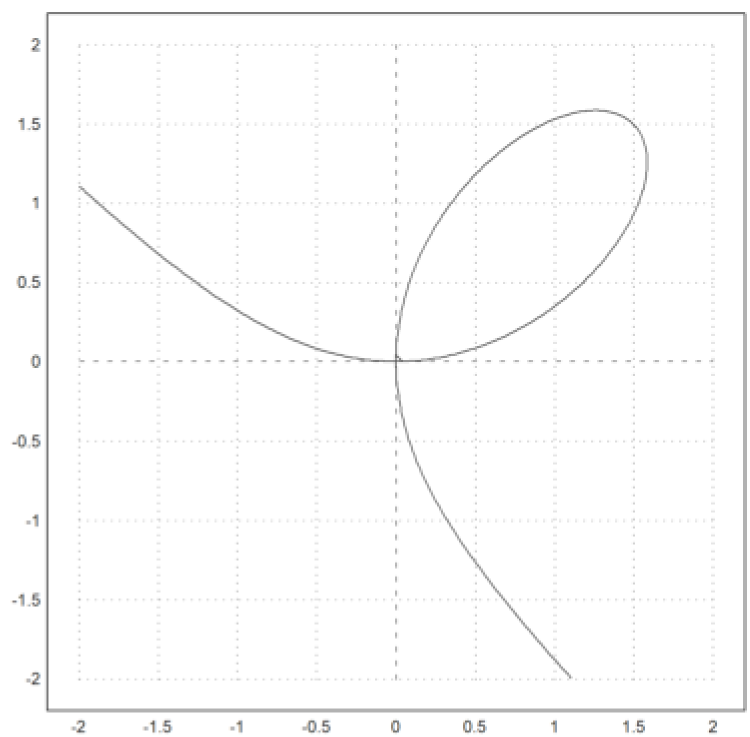
>plot2d(z,a=0,b=2,c=0,d=2,level=[-10;0],n=100,contourwidth=3,style="/"):

>$z with y=l*x, sol &= solve(%,x); $sol

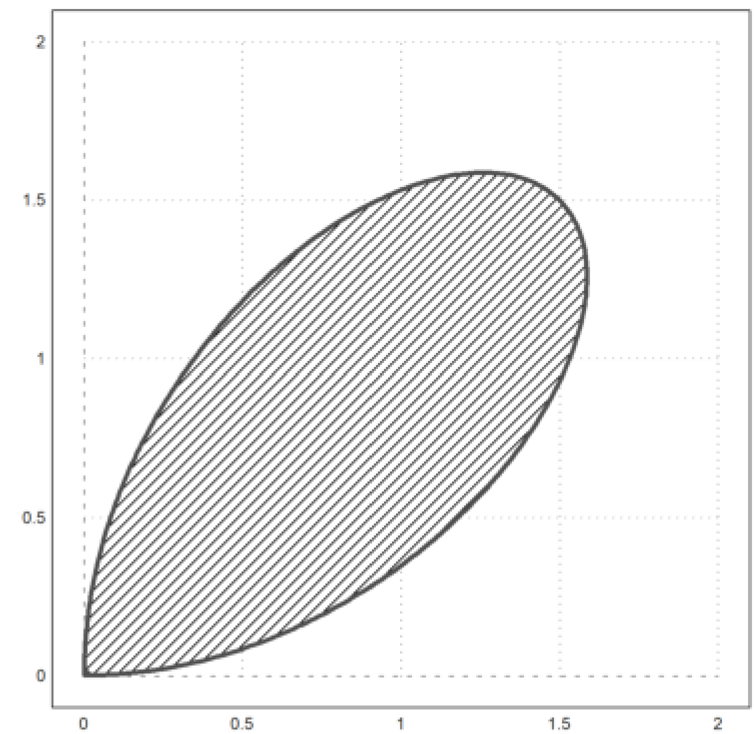
$$l^3\,x^3+x^3-3\,l\,x^2$$



Gambar 366: images/KALKULUS_Diva%20Nagita(23030630024)-250.png



Gambar 367: images/KALKULUS_Diva%20Nagita(23030630024)-253.png



Gambar 368: images/KALKULUS_Diva%20Nagita(23030630024)-254.png

$$\left[x = \frac{3\,l}{l^3+1}, x = 0\right]$$

>function f(l) &= rhs(sol[1]); $'f(l)=f(l)

$$f\left(l\right) = \frac{3\,l}{l^3+1}$$

>plot2d(&f(x),&x*f(x),xmin=-0.5,xmax=2,a=0,b=2,c=0,d=2,r=1.5):

Elemen panjang kurva adalah:

$$ds = \sqrt{f'(l)^2 + (lf'(l) + f(l))^2}.$$

>function ds(l) &= ratsimp(sqrt(diff(f(l),l)$^{2+\text{diff}(l*f(l),l)}$2)); $'ds(l)=ds(l)

$$ds\left(l\right) = \frac{\sqrt{9\,l^8+36\,l^6-36\,l^5-36\,l^3+36\,l^2+9}}{\sqrt{l^{12}+4\,l^9+6\,l^6+4\,l^3+1}}$$

>$integrate(ds(l),l,0,1)

$$\int_{0}^{1}{\frac{\sqrt{9\,l^8+36\,l^6-36\,l^5-36\,l^3+36\,l^2+9}}{\sqrt{l^{12}+4\,l^9+6\,l^6+4\,l^3+1}}\;dl}$$

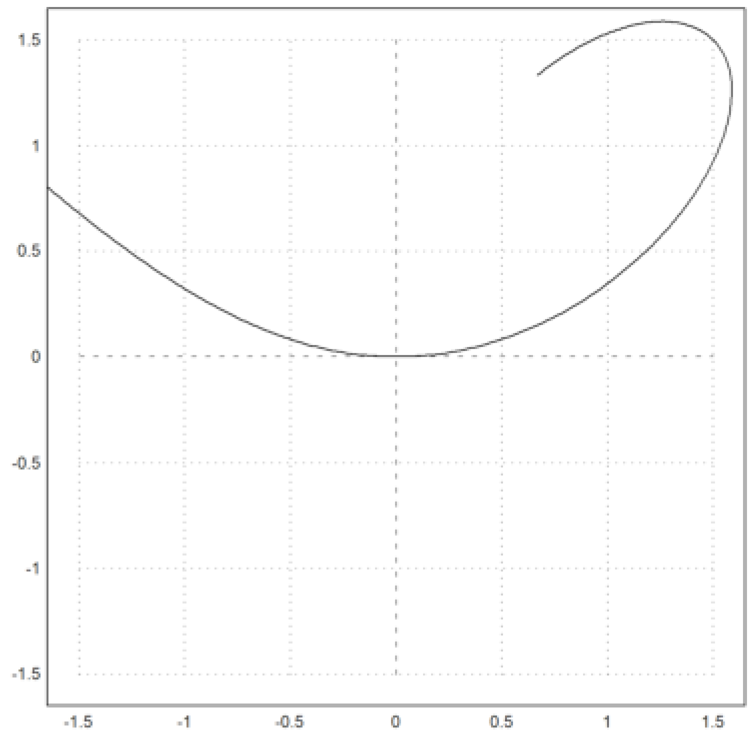
Integral tersebut tidak dapat dihitung secara eksak menggunakan Maxima. Kita hitung integral etrsebut secara numerik dengan Euler. Karena kurva simetris, kita hitung untuk nilai variabel integrasi dari 0 sampai 1, kemudian hasilnya dikalikan 2.

>2*integrate("ds(x)",0,1)

```
4.91748872168
```

>2*romberg(&ds(x),0,1)// perintah Euler lain untuk menghitung nilai hampiran integral yang sama

```
4.91748872168
```



Gambar 369: images/KALKULUS_Diva%20Nagita(23030630024)-258.png

>function panjangkurva(fx,fy,a,b)...

```
ds=mxm("sqrt(diff(@fx,x)^2+diff(@fy,x))^2");
return romberg(ds,a,b);
endfunction
```

>panjangkurva("x","x^2",-1,1)

```
2
```

>2*panjangkurva(mxm("f(x)"),mxm("x*f(x)"),0,1)

```
10.9351907861
```

2. Spiral Logaritmik

$$x = e^{ax}\cos x,\ y = e^{ax}\sin x.$$

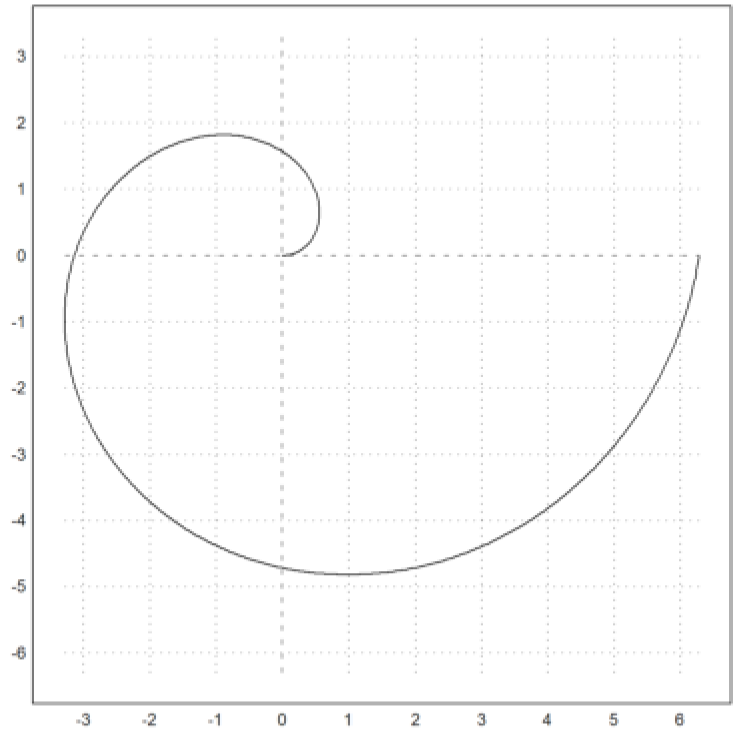
>plot2d("x*cos(x)","x*sin(x)",xmin=0,xmax=2*pi,square=1):
>panjangkurva("x*cos(x)","x*sin(x)",0,2*pi)

```
46.0540912208
```

Berikut kita definisikan fungsi yang sama namun dengan Maxima, untuk perhitungan eksak.
>&kill(ds,x,fx,fy)

```
done
```

>function ds(fx,fy) &&= sqrt(diff(fx,x)$^{2+\text{diff(fy,x)}}$2)



Gambar 370: images/KALKULUS_Diva%20Nagita(23030630024)-263.png

```
                  2                  2
sqrt(diff (fy, x) + diff (fx, x))
```

>sol &= ds(x*cos(x),x*sin(x)); $sol // Kita gunakan untuk menghitung panjang kurva terakhir di atas

$$\sqrt{\left(\cos x-x\,\sin x\right)^2+\left(\sin x+x\,\cos x\right)^2}$$

>$sol | trigreduce | expand, $integrate(%,x,0,2*pi), %()

$$\sqrt{x^2+1}$$

$$\frac{\operatorname{asinh}\left(2\,\pi\right)+2\,\pi\,\sqrt{4\,\pi^2+1}}{2}$$

21.2562941482

>plot2d(“3*x$^{2-1”,”3*x}$3-1”,xmin=-1/sqrt(3),xmax=1/sqrt(3),square=1):

>sol &= radcan(ds(3*x$^{2-1,3*x}$3-1)); $sol

$$3\,x\,\sqrt{9\,x^2+4}$$

>$showev('integrate(sol,x,0,1/sqrt(3))), $2*float(%) // panjang kurva di atas

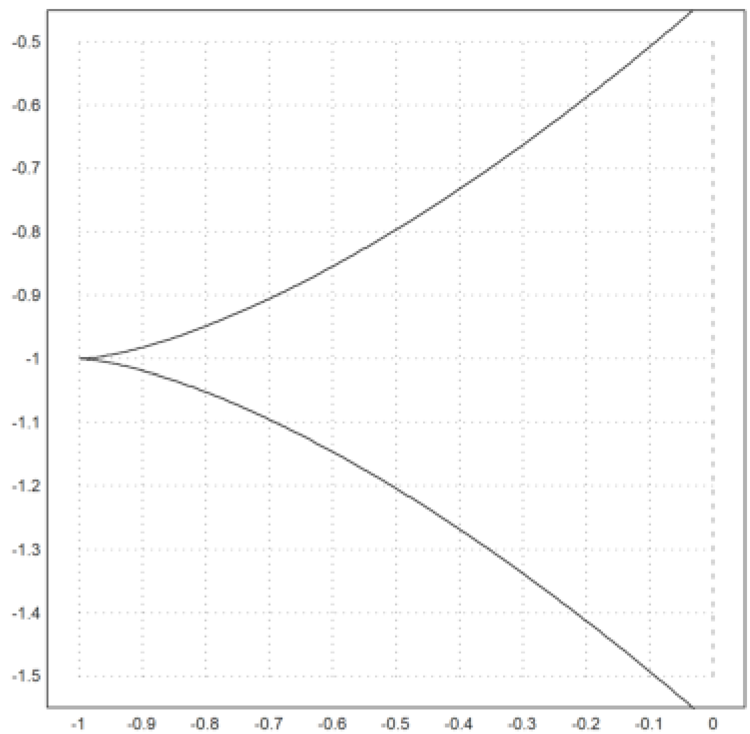
$$3\,\int_{0}^{\frac{1}{\sqrt{3}}}{x\,\sqrt{9\,x^2+4}\;dx}=3\,\left(\frac{7^{\frac{3}{2}}}{27}-\frac{8}{27}\right)$$

$$6.0\,\int_{0.0}^{0.5773502691896258}{x\,\sqrt{9.0\,x^2+4.0}\;dx}=2.337835372767141$$

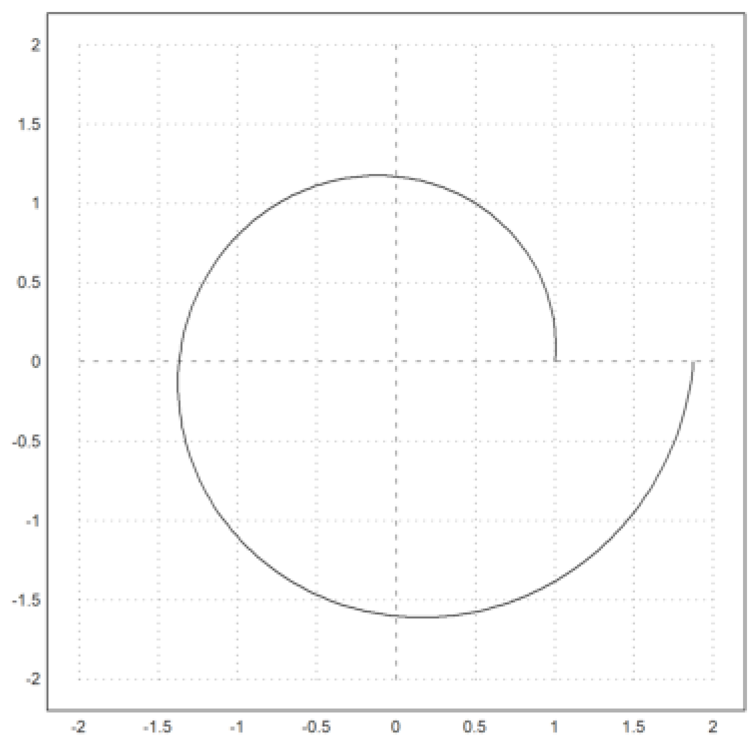
>a=0.1; plot2d(“exp(a*x)*cos(x)”,“exp(a*x)*sin(x)”,r=2,xmin=0,xmax=2*pi):

>&kill(a) // hapus expresi a

```
done
```



Gambar 371: images/KALKULUS_Diva%20Nagita(23030630024)-267.png



Gambar 372: images/KALKULUS_Diva%20Nagita(23030630024)-271.png

>function fx(t) &= exp(a*t)*cos(t); $'fx(t)=fx(t)

$$fx\left(t\right)=e^{a\,t}\,\cos t$$

>function fy(t) &= exp(a*t)*sin(t); $'fy(t)=fy(t)

$$fy\left(t\right)=e^{a\,t}\,\sin t$$

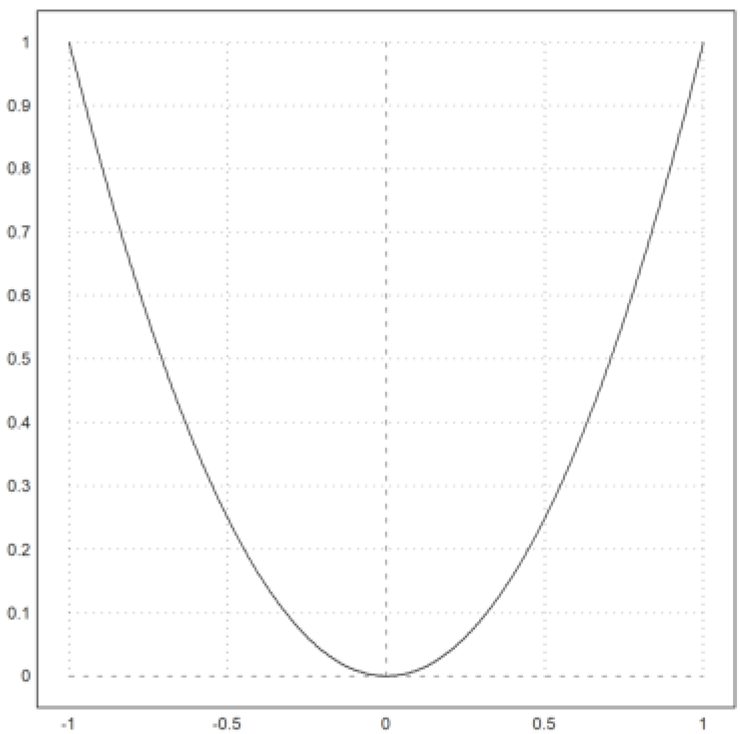
>function df(t) &= trigreduce(radcan(sqrt(diff(fx(t),t)$^{2+\text{diff(fy(t),t)}}$2))); $'df(t)=df(t)
>S &=integrate(df(t),t,0,2*%pi); $S

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

>S(a=0.1) // Panjang kurva untuk a=0.1

```
8.78817491636
```

>plot2d("x^2",xmin=-1,xmax=1):



Gambar 373: images/KALKULUS_Diva%20Nagita(23030630024)-275.png

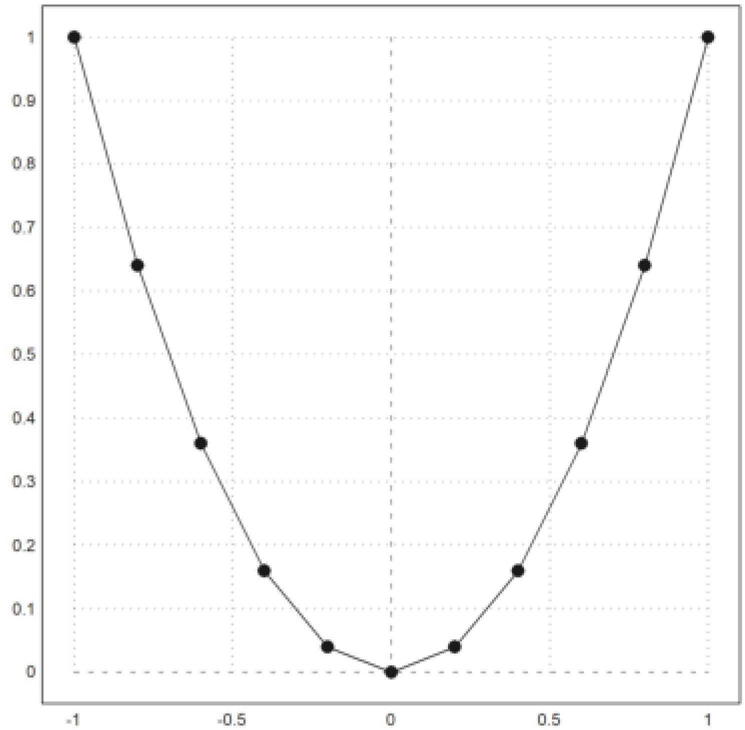
>$showev('integrate(sqrt(1+diff(x^{2,x})2),x,-1,1))

$$\int_{-1}^{1}{\sqrt{4\,x^2+1}\;dx}=\frac{\operatorname{asinh}2+2\,\sqrt{5}}{2}$$

>$float(%)

$$\int_{-1.0}^{1.0}{\sqrt{4.0\,x^2+1.0}\;dx}=2.957885715089195$$

>x=-1:0.2:1; y=x^2; plot2d(x,y); ...
> plot2d(x,y,points=1,style="o#",add=1):
>i=1:cols(x)-1; sum(sqrt((x[i+1]-x[i])$^{2+\text{(y[i+1]-y[i])}}$2))



Gambar 374: images/KALKULUS_Diva%20Nagita(23030630024)-278.png

```
2.95191957027
```

Sikloid

Berikut kita akan menghitung panjang kurva lintasan (sikloid) suatu titik pada lingkaran yang berputar ke kanan pada permukaan datar. Misalkan jari-jari lingkaran tersebut adalah r. Posisi titik pusat lingkaran pada saat t adalah:

$$(rt, r).$$

Misalkan posisi titik pada lingkaran tersebut mula-mula (0,0) dan posisinya pada saat t adalah:

$$(r(t - \sin(t)), r(1 - \cos(t))).$$

Berikut kita plot lintasan tersebut dan beberapa posisi lingkaran ketika t=0, t=pi/2, t=r*pi.

>x &= r*(t-sin(t))

```
        [0, 1.665833531718508e-4 r, 0.001330669204938795 r,
0.004479793338660443 r, 0.0105816576913495 r, 0.02057446139579699 r,
0.03535752660496461 r, 0.05578231276230894 r, 0.08264390910047725 r,
0.1166730903725166 r, 0.1585290151921035 r, 0.2087926399385646 r,
0.2679609140327737 r, 0.3364418145828071 r, 0.4145502700115399 r,
0.5025050133959458 r, 0.6004263969584952 r, 0.7083351895475318 r,
0.8261523691218055 r, 0.9536999123125863 r, 1.090702573174319 r,
1.236790633351127 r, 1.391503596180411 r, 1.554294787823281 r,
1.724536819448851 r, 1.901527855896045 r, 2.084498628178538 r,
2.272620119766172 r, 2.465011849844097 r, 2.66075067078602 r,
2.858879991940135 r]
```

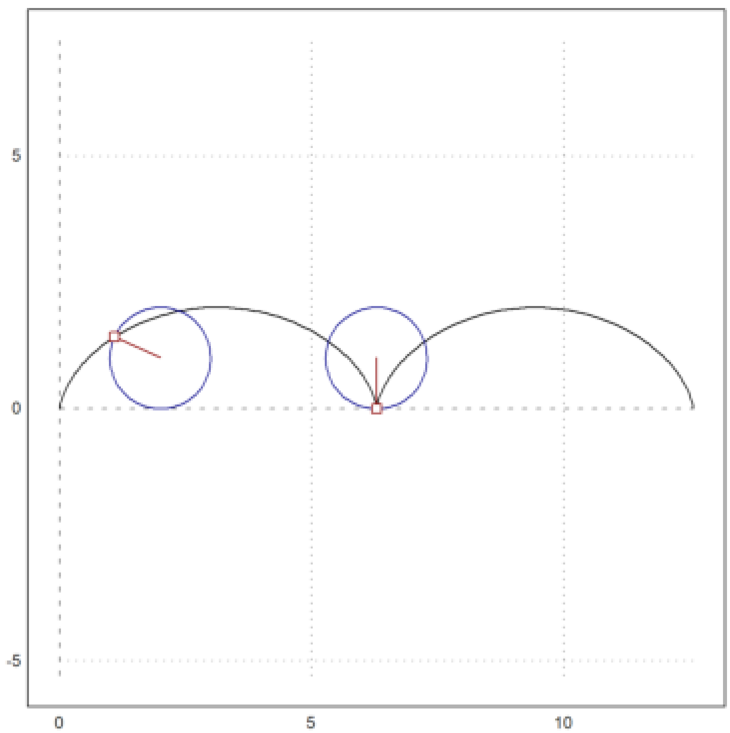
>y &= r*(1-cos(t))

```
        [0, 0.004995834721974179 r, 0.01993342215875837 r,
0.04466351087439402 r, 0.0789390059971149 r, 0.1224174381096272 r,
0.1746643850903217 r, 0.2351578127155115 r, 0.3032932906528345 r,
0.3783900317293355 r, 0.4596976941318602 r, 0.5464038785744225 r,
0.6376422455233264 r, 0.7325011713754126 r, 0.8300328570997592 r,
0.9292627983322973 r, 1.029199522301289 r, 1.128844494295525 r,
1.227202094693087 r, 1.323289566863504 r, 1.416146836547143 r,
1.504846104599858 r, 1.588501117255346 r, 1.666276021279825 r,
1.737393715541246 r, 1.801143615546934 r, 1.856888753368948 r,
1.904072142017062 r, 1.942222340668659 r, 1.970958165149591 r,
1.989992496600446 r]
```

Berikut kita gambar sikloid untuk r=1.

>ex &= x-sin(x); ey &= 1-cos(x); aspect(1);

>plot2d(ex,ey,xmin=0,xmax=4pi,square=1); . . .
> plot2d("2+cos(x)","1+sin(x)",xmin=0,xmax=2pi,>add,color=blue); . . .
> plot2d([2,ex(2)],[1,ey(2)],color=red,>add); . . .
> plot2d(ex(2),ey(2),>points,>add,color=red); . . .
> plot2d("2pi+cos(x)","1+sin(x)",xmin=0,xmax=2pi,>add,color=blue); . . .
> plot2d([2pi,ex(2pi)],[1,ey(2pi)],color=red,>add); . . .
> plot2d(ex(2pi),ey(2pi),>points,>add,color=red):



Gambar 375: images/KALKULUS_Diva%20Nagita(23030630024)-281.png

>ds &= radcan(sqrt(diff(ex,x)^{2+diff(ey,x)}2)); $ds=trigsimp(ds) // elemen panjang kurva sikloid

$$\sqrt{\sin^2 x + \cos^2 x - 2\,\cos x + 1} = \sqrt{2 - 2\,\cos x}$$

>ds &= trigsimp(ds); $ds

$$\sqrt{2 - 2\,\cos x}$$

>$showev('integrate(ds,x,0,2*pi)) // hitung panjang sikloid satu putaran penuh

$$\int_{0}^{2\,\pi}{\sqrt{2-2\,\cos x}\;dx}=8$$

>integrate(mxm("ds"),0,2*pi) // hitung secara numerik

```
8
```

>romberg(mxm("ds"),0,2*pi) // cara lain hitung secara numerik

```
8
```

Kurvatur (Kelengkungan) Kurva

image: Osculating.png

Aslinya, kelengkungan kurva diferensiabel (yakni, kurva mulus yang tidak lancip) di titik P didefinisikan melalui lingkaran oskulasi (yaitu, lingkaran yang melalui titik P dan terbaik memperkirakan, paling banyak menyinggung kurva di sekitar P). Pusat dan radius kelengkungan kurva di P adalah pusat dan radius lingkaran oskulasi. Kelengkungan adalah kebalikan dari radius kelengkungan:

$$\kappa = \frac{1}{R}$$

dengan R adalah radius kelengkungan. (Setiap lingkaran memiliki kelengkungan ini pada setiap titiknya, dapat diartikan, setiap lingkaran berputar 2pi sejauh 2piR.)

Definisi ini sulit dimanipulasi dan dinyatakan ke dalam rumus untuk kurva umum. Oleh karena itu digunakan definisi lain yang ekivalen.

Definisi Kurvatur dengan Fungsi Parametrik Panjang Kurva

Setiap kurva diferensiabel dapat dinyatakan dengan persamaan parametrik terhadap panjang kurva s:

$$\gamma(s) = (x(s),\ y(s)),$$

dengan x dan y adalah fungsi riil yang diferensiabel, yang memenuhi:

$$\|\gamma'(s)\| = \sqrt{x'(s)^2 + y'(s)^2} = 1.$$

Ini berarti bahwa vektor singgung

$$\mathbf{T}(s) = (x'(s),\ y'(s))$$

memiliki norm 1 dan merupakan vektor singgung satuan.

Apabila kurvanya memiliki turunan kedua, artinya turunan kedua x dan y ada, maka T'(s) ada. Vektor ini merupakan normal kurva yang arahnya menuju pusat kurvatur, norm-nya merupakan nilai kurvatur (kelengkungan):

$$\begin{aligned}\mathbf{T}(s) &= \gamma'(s),\\ \mathbf{T}^2(s) &= 1 \text{ (konstanta)} \Rightarrow \mathbf{T}'(s)\cdot\mathbf{T}(s) = 0\\ \kappa(s) &= \|\mathbf{T}'(s)\| = \|\gamma''(s)\| = \sqrt{x''(s)^2 + y''(s)^2}.\end{aligned}$$

Nilai

$$R(s) = \frac{1}{\kappa(s)}$$

disebut jari-jari (radius) kelengkungan kurva.

Bilangan riil

$$k(s) = \pm\kappa(s)$$

disebut nilai kelengkungan bertanda.

Contoh:

Akan ditentukan kurvatur lingkaran

$$x = r\cos t,\ y = r\sin t.$$

>fx &= r*cos(t); fy &=r*sin(t);

>&assume(t>0,r>0); s &=integrate(sqrt(diff(fx,t)$^{2+\text{diff(fy,t)}}$2),t,0,t); s // elemen panjang kurva, panjang busur lingkaran (s)

```
r t
```

>&kill(s); fx &= r*cos(s/r); fy &=r*sin(s/r); // definisi ulang persamaan parametrik terhadap s dengan substitusi t=s/r

>k &= trigsimp(sqrt(diff(fx,s,2)$^{2+\text{diff(fy,s,2)}}$2)); $k // nilai kurvatur lingkaran dengan menggunakan definisi di atas

$$\frac{1}{r}$$

Untuk representasi parametrik umum, misalkan

$$x = x(t),\ y = y(t)$$

merupakan persamaan parametrik untuk kurva bidang yang terdiferensialkan dua kali. Kurvatur untuk kurva tersebut didefinisikan sebagai

$$\begin{aligned}\kappa &= \frac{d\phi}{ds} = \frac{\frac{d\phi}{dt}}{\frac{ds}{dt}} \quad (\phi \text{ adalah sudut kemiringan garis singgung dan } s \text{ adalah panjang kurva}) \\ &= \frac{\frac{d\phi}{dt}}{\sqrt{(\frac{dx}{dt})^2 + (\frac{dy}{dt})^2}} = \frac{\frac{d\phi}{dt}}{\sqrt{x'(t)^2 + y'(t)^2}}.\end{aligned}$$

Selanjutnya, pembilang pada persamaan di atas dapat dicari sebagai berikut.

$$\sec^2\phi\frac{d\phi}{dt} = \frac{d}{dt}(\tan\phi) = \frac{d}{dt}\left(\frac{dy}{dx}\right) = \frac{d}{dt}\left(\frac{dy/dt}{dx/dt}\right) = \frac{d}{dt}\left(\frac{y'(t)}{x'(t)}\right) = \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2}.$$

$$\begin{aligned}\frac{d\phi}{dt} &= \frac{1}{\sec^2\phi}\frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2} \\ &= \frac{1}{1+\tan^2\phi}\frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2} \\ &= \frac{1}{1+\left(\frac{y'(t)}{x'(t)}\right)^2}\frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2} \\ &= \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2 + y'(t)^2}.\end{aligned}$$

Jadi, rumus kurvatur untuk kurva parametrik

$$x = x(t),\ y = y(t)$$

adalah

$$\kappa(t)=\frac{x'(t)y''(t)-x''(t)y'(t)}{(x'(t)^2+y'(t)^2)^{3/2}}.$$

Jika kurvanya dinyatakan dengan persamaan parametrik pada koordinat kutub

$$x=r(\theta)\cos\theta,\ y=r(\theta)\sin\theta,$$

maka rumus kurvaturnya adalah

$$\kappa(\theta)=\frac{r(\theta)^2+2r'(\theta)^2-r(\theta)r''(\theta)}{(r'(\theta)^2+r'(\theta)^2)^{3/2}}.$$

(Silakan Anda turunkan rumus tersebut!)
Contoh:
Lingkaran dengan pusat (0,0) dan jari-jari r dapat dinyatakan dengan persamaan parametrik

$$x=r\cos t,\ y=r\sin t.$$

Nilai kelengkungan lingkaran tersebut adalah

$$\kappa(t)=\frac{x'(t)y''(t)-x''(t)y'(t)}{(x'(t)^2+y'(t)^2)^{3/2}}=\frac{r^2}{r^3}=\frac{1}{r}.$$

Hasil cocok dengan definisi kurvatur suatu kelengkungan.
Kurva

$$y=f(x)$$

dapat dinyatakan ke dalam persamaan parametrik

$$x=t,\ y=f(t),\ \text{ dengan } x'(t)=1,\ x''(t)=0,$$

sehingga kurvaturnya adalah

$$\kappa(t)=\frac{y''(t)}{(1+y'(t)^2)^{3/2}}.$$

Contoh:
Akan ditentukan kurvatur parabola

$$y=ax^2+bx+c.$$

>function f(x) &= a*x^2+b*x+c; $y=f(x)

$$y=a\,x^2+b\,x+c$$

>function k(x) &= (diff(f(x),x,2))/(1+diff(f(x),x)$^{2)}$(3/2); $'k(x)=k(x) // kelengkungan parabola

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

>function f(x) &= x^2+x+1; $y=f(x) // akan kita plot kelengkungan parabola untuk a=b=c=1

$$y=x^2+x+1$$

>function k(x) &= (diff(f(x),x,2))/(1+diff(f(x),x)$^{2)}$(3/2); $'k(x)=k(x) // kelengkungan parabola

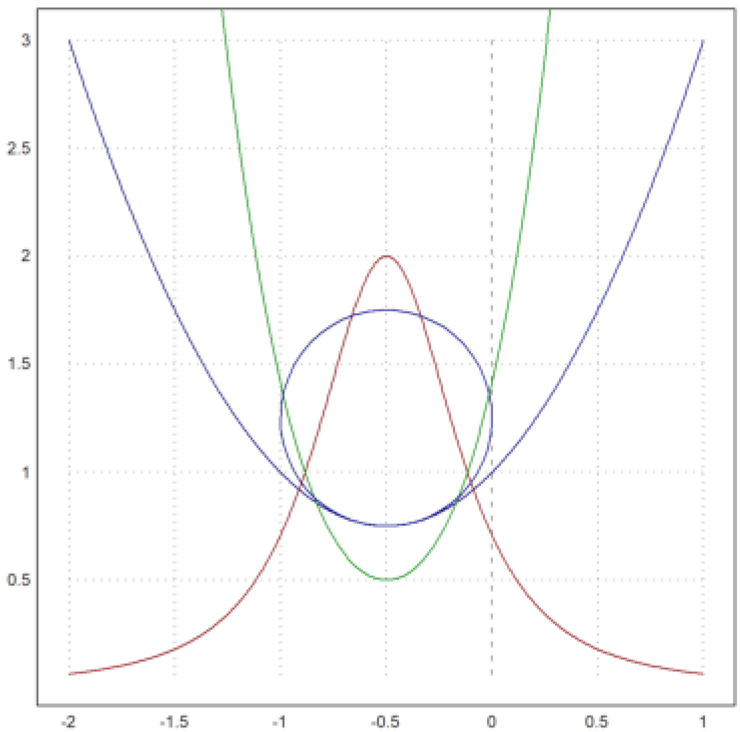
$$k\left(x\right)=\frac{2}{\left(\left(2\,x+1\right)^2+1\right)^{\frac{3}{2}}}$$

Berikut kita gambar parabola tersebut beserta kurva kelengkungan, kurva jari-jari kelengkungan dan salah satu lingkaran oskulasi di titik puncak parabola. Perhatikan, puncak parabola dan jari-jari lingkaran oskulasi di puncak parabola adalah

$$(-1/2, 3/4),\ 1/k(2) = 1/2,$$

sehingga pusat lingkaran oskulasi adalah (-1/2, 5/4).

>plot2d(["f(x)", "k(x)"],-2,1, color=[blue,red]); plot2d("1/k(x)",-1.5,1,color=green,>add); ...
> plot2d("-1/2+1/k(-1/2)*cos(x)","5/4+1/k(-1/2)*sin(x)",xmin=0,xmax=2pi,>add,color=blue):



Gambar 376: images/KALKULUS_Diva%20Nagita(23030630024)-312.png

Untuk kurva yang dinyatakan dengan fungsi implisit

$$F(x, y) = 0$$

dengan turunan-turunan parsial

$$F_x = \frac{\partial F}{\partial x},\ F_y = \frac{\partial F}{\partial y},\ F_{xy} = \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial x}\right),\ F_{xx} = \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial x}\right),\ F_{yy} = \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial y}\right),$$

berlaku

$$F_x dx + F_y dy = 0 \text{ atau } \frac{dy}{dx} = -\frac{F_x}{F_y},$$

sehingga kurvaturnya adalah

$$\kappa = \frac{F_y^2 F_{xx} - 2F_x F_y F_{xy} + F_x^2 F_{yy}}{\left(F_x^2 + F_y^2\right)^{3/2}}.$$

(Silakan Anda turunkan sendiri!)

Contoh 1:

Parabola

$$y = ax^2 + bx + c$$

dapat dinyatakan ke dalam persamaan implisit

$$ax^2 + bx + c - y = 0.$$

>function F(x,y) &=a*x^2+b*x+c-y; $F(x,y)

$$-y + a\,x^2 + b\,x + c$$

>Fx &= diff(F(x,y),x), Fxx &=diff(F(x,y),x,2), Fy &=diff(F(x,y),y), Fxy &=diff(diff(F(x,y),x),y), Fyy &=diff(F(x,y),y,2)

```
2 a x + b


2 a


- 1


0


0
```

>function k(x) &= (Fy$^{2*Fxx-2*Fx*Fy*Fxy+Fx}$2*Fyy)/(Fx^{2+Fy}2)^(3/2); $'k(x)=k(x) // kurvatur parabola tersebut

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

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: + dengan cara yang sama seperti mendefinisikan vektor dengan elemen-elemen beraturan (menggunakan titik dua ":"); + menggunakan perintah "sequence" dan rumus barisan (suku ke -n); + menggunakan perintah "iterate" atau "niterate"; + menggunakan fungsi Maxima "create_list" atau "makelist" untuk - menghasilkan barisan simbolik; - menggunakan fungsi biasa yang inputnya vektor atau barisan; - menggunakan fungsi rekursif.

EMT menyediakan beberapa perintah (fungsi) terkait barisan, yakni:

- sum: menghitung jumlah semua elemen suatu barisan
- cumsum: jumlah kumulatif suatu barisan
- differences: selisih antar elemen-elemen berturutan

Contoh:

>5:15

```
[5,  6,  7,  8,  9,  10,  11,  12,  13,  14,  15]
```

>1:3:10

```
[1,  4,  7,  10]
```

>sum(1:3:10)

```
22
```

Rumus Deret Berhingga

Diberikan deret sederhana sebagai berikut
>A := [2, 4, 8, 16, 32]

```
[2,  4,  8,  16,  32]
```

Menentukan suku ke-n

$$a_n = ar^{n-1}$$

Contoh:
Tentukanlah suku ke-7 dari deret tersebut
Dengan a=2 dan r=2
>n=7; a=2; r=2; a*r^(n-1)

```
128
```

Menghitung Jumlah deret suku ke-n

$$S_n = \frac{a(r^n - 1)}{r - 1}$$

jika konvergen, r > 1
atau

$$S_n = \frac{a(1 - r^n)}{1 - r}$$

jika divergen, r < 1
Contoh:
Hitunglah jumlah deret A pada suku ke-5
Dengan a=2 dan r=2
>n=5; a=2; r=2; a*((r^n)-1)/(r-1)

```
62
```

>sum(A := [2, 4, 8, 16, 32])

```
62
```

Rumus Deret Tak Hingga

Menghitung jumlah suku tak hingga konvergen
Untuk -1 < r < 1, maka

$$S_\infty = U_1 + U_2 + U_3 + ...$$

Memiliki limit jumlah

$$S_\infty = \frac{a}{1 - r}$$

dengan

$$U_1 = a$$

Contoh:

$$S_\infty = 12 + 6 + 3 + ...$$

>a=12; r=1/2; a/(1-r)

```
24
```

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))

$$\lim_{n \to \infty} \frac{1}{n} = 0$$

2.

$$\lim_{n \to \infty} \frac{2n^2 - 1}{n^2 + 5}$$

>$showev('limit((2*n$^{2-1)/(n}$2+5),n,inf))

$$\lim_{n \to \infty} \frac{2\,n^2 - 1}{n^2 + 5} = 2$$

Iterasi dan Barisan

EMT menyediakan fungsi iterate("g(x)", x0, n) untuk melakukan iterasi

$$x_{k+1} = g(x_k),\ x_0 = x_0, k = 1, 2, 3, ..., n.$$

Berikut ini disajikan contoh-contoh penggunaan iterasi dan rekursi dengan EMT. Contoh pertama menunjukkan pertumbuhan dari nilai awal 1000 dengan laju pertambahan 5%, selama 10 periode.

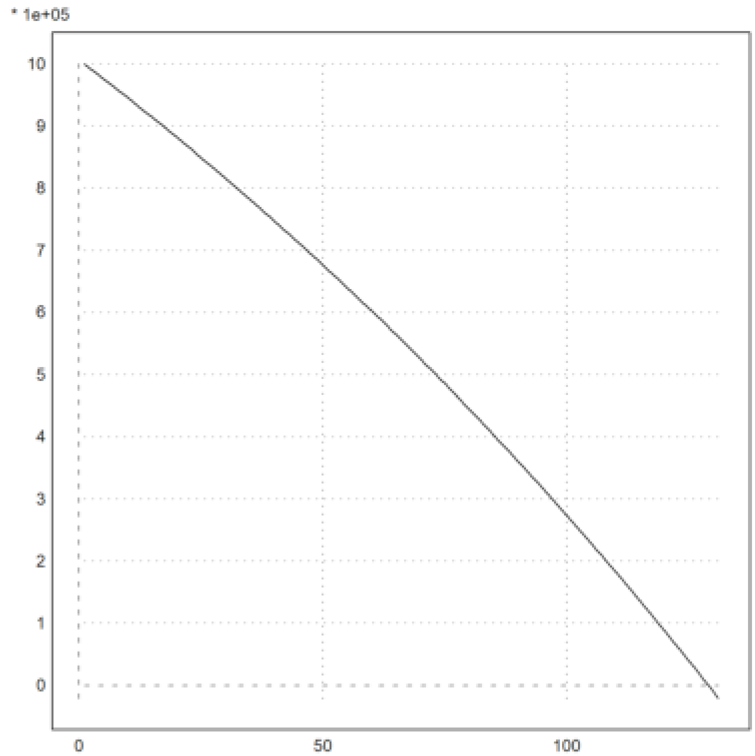
>q=1.05; iterate("x*q",1000,n=10)'

```
      1000
      1050
    1102.5
   1157.63
   1215.51
   1276.28
```

```
 1340.1
 1407.1
1477.46
1551.33
1628.89
```

Contoh berikutnya memperlihatkan bahaya menabung di bank pada masa sekarang! Dengan bunga tabungan sebesar 6% per tahun atau 0.5% per bulan dipotong pajak 20%, dan biaya administrasi 10000 per bulan, tabungan sebesar 1 juta tanpa diambil selama sekitar 10 tahunan akan habis diambil oleh bank!

>r=0.005; plot2d(iterate("(1+0.8*r)*x-10000",1000000,n=130)):



Gambar 377: images/KALKULUS_Diva%20Nagita(23030630024)-335.png

Silakan Anda coba-coba, dengan tabungan minimal berapa agar tidak akan habis diambil oleh bank dengan ketentuan bunga dan biaya administrasi seperti di atas.

Berikut adalah perhitungan minimal tabungan agar aman di bank dengan bunga sebesar r dan biaya administrasi a, pajak bunga 20%.

>$solve(0.8*r*A-a,A), $% with [r=0.005, a=10]

$$[A = 2500.0]$$

$[A = 2500.0]$

Berikut didefinisikan fungsi untuk menghitung saldo tabungan, kemudian dilakukan iterasi.

>function saldo(x,r,a) := round((1+0.8*r)*x-a,2);

>iterate({{"saldo",0.005,10}},1000,n=6)

```
[1000,  994,  987.98,  981.93,  975.86,  969.76,  963.64]
```

>iterate({{"saldo",0.005,10}},2000,n=6)

```
[2000,  1998,  1995.99,  1993.97,  1991.95,  1989.92,  1987.88]
```

>iterate({{"saldo",0.005,10}},2500,n=6)

```
[2500,  2500,  2500,  2500,  2500,  2500,  2500]
```

Tabungan senilai 2,5 juta akan aman dan tidak akan berubah nilai (jika tidak ada penarikan), sedangkan jika tabungan awal kurang dari 2,5 juta, lama kelamaan akan berkurang meskipun tidak pernah dilakukan penarikan uang tabungan.

>iterate({{"saldo",0.005,10}},3000,n=6)

```
[3000,  3002,  3004.01,  3006.03,  3008.05,  3010.08,  3012.12]
```

Tabungan yang lebih dari 2,5 juta baru akan bertambah jika tidak ada penarikan.

Untuk barisan yang lebih kompleks dapat digunakan fungsi "sequence()". Fungsi ini menghitung nilai-nilai x[n] dari semua nilai sebelumnya, x[1],...,x[n-1] yang diketahui.

Berikut adalah contoh barisan Fibonacci.

$$x_n = x_{n-1} + x_{n-2}, \quad x_1 = 1, \quad x_2 = 1$$

>sequence("x[n-1]+x[n-2]",[1,1],15)

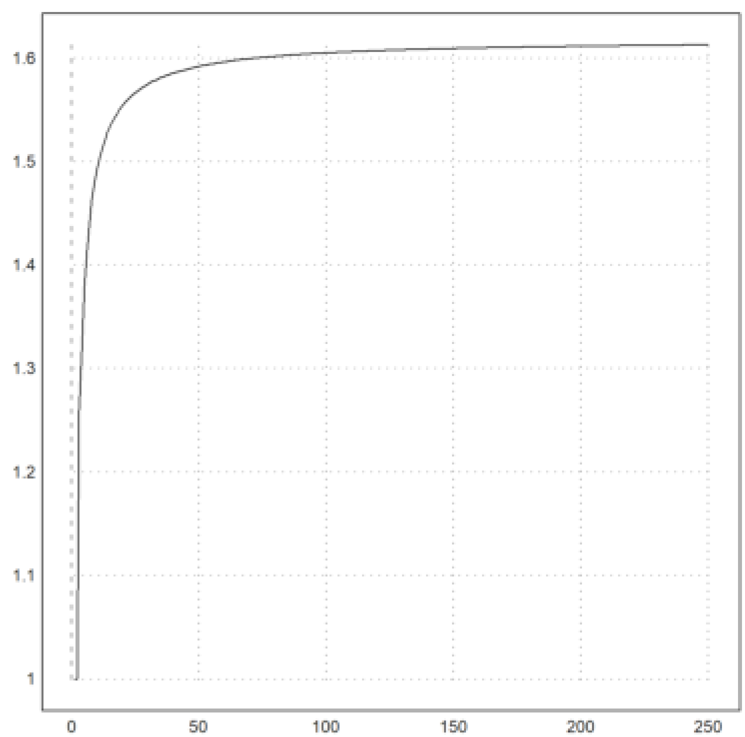
```
[1,  1,  2,  3,  5,  8,  13,  21,  34,  55,  89,  144,  233,  377,  610]
```

Barisan Fibonacci memiliki banyak sifat menarik, salah satunya adalah akar pangkat ke-n suku ke-n akan konvergen ke pecahan emas:

>$'(1+sqrt(5))/2=float((1+sqrt(5))/2)

$$\frac{\sqrt{5}+1}{2} = 1.618033988749895$$

>plot2d(sequence("x[n-1]+x[n-2]",[1,1],250)^(1/(1:250))):



Gambar 378: images/KALKULUS_Diva%20Nagita(23030630024)-340.png

Barisan yang sama juga dapat dihasilkan dengan menggunakan loop.

>x=ones(500); for k=3 to 500; x[k]=x[k-1]+x[k-2]; end;

Rekursi dapat dilakukan dengan menggunakan rumus yang tergantung pada semua elemen sebelumnya. Pada contoh berikut, elemen ke-n merupakan jumlah (n-1) elemen sebelumnya, dimulai dengan 1 (elemen ke-1). Jelas, nilai elemen ke-n adalah 2ˆ(n-2), untuk n=2, 4, 5, ...

>sequence("sum(x)",1,10)

```
[1,  1,  2,  4,  8,  16,  32,  64,  128,  256]
```

Selain menggunakan ekspresi dalam x dan n, kita juga dapat menggunakan fungsi.

Pada contoh berikut, digunakan iterasi

$$x_n = A \cdot x_{n-1},$$

dengan A suatu matriks 2x2, dan setiap x[n] merupakan matriks/vektor 2x1.

>A=[1,1;1,2]; function suku(x,n) := A.x[,n-1]

>sequence("suku",[1;1],6)

```
Real 2 x 6 matrix

            1            2            5           13      ...
            1            3            8           21      ...
```

Hasil yang sama juga dapat diperoleh dengan menggunakan fungsi perpangkatan matriks "matrixpower()". Cara ini lebih cepat, karena hanya menggunakan perkalian matriks sebanyak log_2(n).

$$x_n = A.x_{n-1} = A^2.x_{n-2} = A^3.x_{n-3} = ... = A^{n-1}.x_1.$$

>sequence("matrixpower(A,n).[1;1]",1,6)

```
Real 2 x 6 matrix

            1            5           13           34      ...
            1            8           21           55      ...
```

Spiral Theodorus

image: Spiral_of_Theodorus.png

Spiral Theodorus (spiral segitiga siku-siku) dapat digambar secara rekursif. Rumus rekursifnya adalah:

$$x_n = \left(1 + \frac{i}{\sqrt{n-1}}\right) x_{n-1}, \quad x_1 = 1,$$

yang menghasilkan barisan bilangan kompleks.

>function g(n) := 1+I/sqrt(n)

Rekursinya dapat dijalankan sebanyak 17 untuk menghasilkan barisan 17 bilangan kompleks, kemudian digambar bilangan-bilangan kompleksnya.

>x=sequence("g(n-1)*x[n-1]",1,17); plot2d(x,r=3.5); textbox(latex("Spiral\ Theodorus"),0.4):

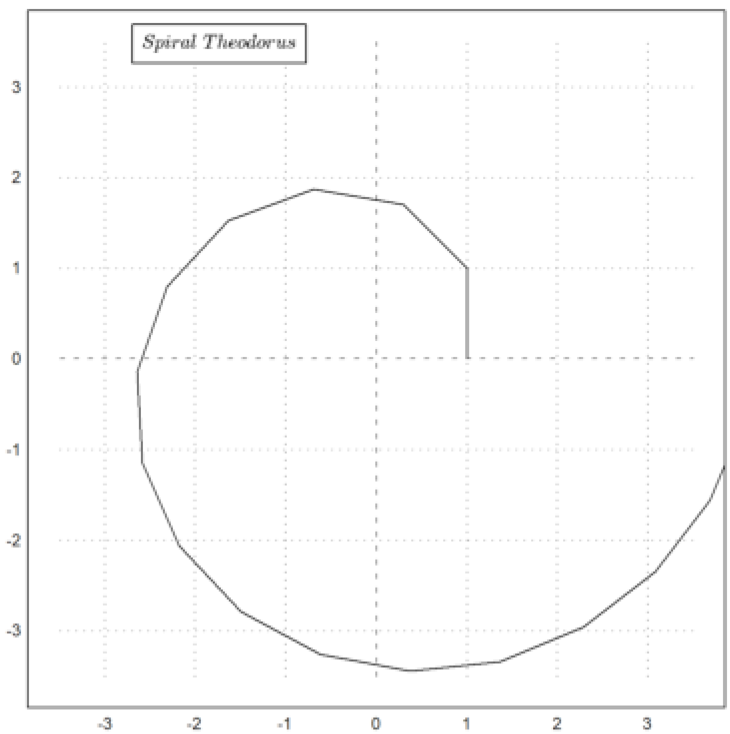
Selanjutnya dihubungan titik 0 dengan titik-titik kompleks tersebut menggunakan loop.

>for i=1:cols(x); plot2d([0,x[i]],>add); end:

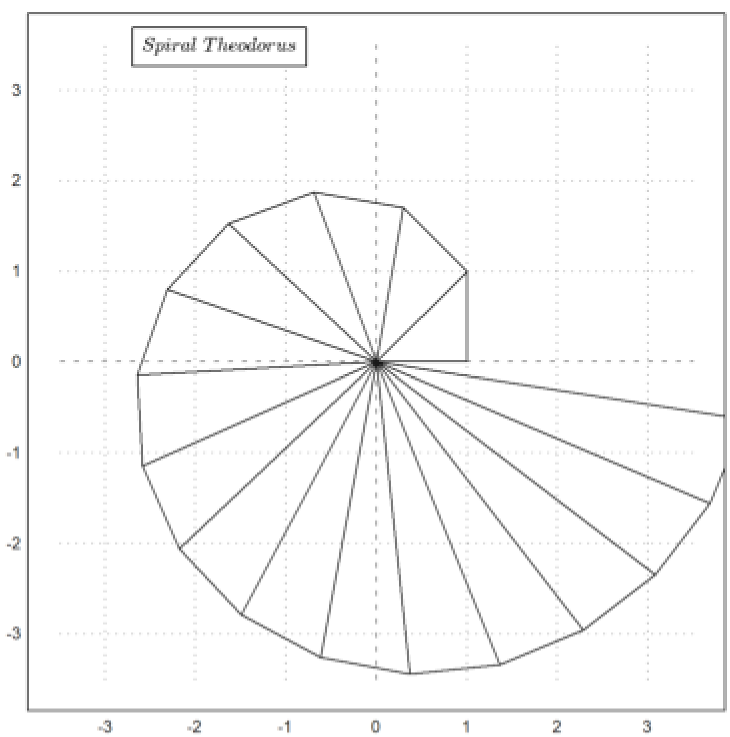
>

Spiral tersebut juga dapat didefinisikan menggunakan fungsi rekursif, yang tidak memmerlukan indeks dan bilangan kompleks. Dalam hal ini diigunakan vektor kolom pada bidang.

>function gstep (v) . . .



Gambar 379: images/KALKULUS_Diva%20Nagita(23030630024)-344.png

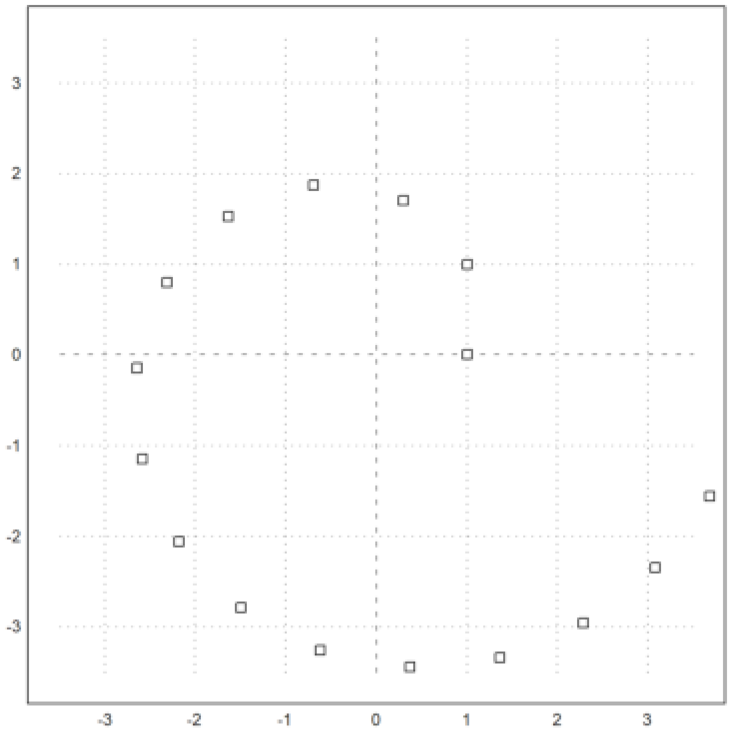


Gambar 380: images/KALKULUS_Diva%20Nagita(23030630024)-345.png

```
w=[-v[2];v[1]];
return v+w/norm(w);
endfunction
```

Jika dilakukan iterasi 16 kali dimulai dari [1;0] akan didapatkan matriks yang memuat vektor-vektor dari setiap iterasi.

>x=iterate("gstep",[1;0],16); plot2d(x[1],x[2],r=3.5,>points):



Gambar 381: images/KALKULUS_Diva%20Nagita(23030630024)-346.png

Kekonvergenan

Barisan yang konvergen adalah barisan yang terbatas.

Jika (Xn) konvergen, maka (Xn) terbatas.

Kita dapat menggunakan fungsi iterate ("g(x)", x0, n) untuk melakukan iterasi

$$x_{k+1} = g(x_k),\ x_0 = x_0, k = 1, 2, 3, ..., n.$$

Contoh:

>iterate("cos(x)",1) // iterasi x(n+1)=cos(x(n)), dengan x(0)=1.

```
0.739085133216
```

Iterasi tersebut konvergen ke penyelesaian persamaan

$$x = \cos(x).$$

Iterasi ini juga dapat dilakukan pada interval, hasilnya adalah barisan interval yang memuat akar tersebut.

>hasil := iterate("cos(x)",2,3) //interval awal (2,3)

```
~0.7390851332143,0.7390851332161~
```

Jika interval hasil tersebut sedikit diperlebar, akan terlihat bahwa interval tersebut memuat akar persamaan x=cos(x).

>h=expand(hasil,100), cos(h) « h

```
~0.73908513313,0.7390851333~
1
```

Iterasi juga dapat digunakan pada fungsi yang didefinisikan.

>function f(x) := (x+4/x)/4

Iterasi x(n+1)=f(x(n)) akan konvergen ke akar kuadrat 4.

>iterate("f",4), sqrt(4)

```
1.15470053838
2
```

>iterate("f",2,5)

```
[2,  1,  1.25,  1.1125,  1.177,  1.14387]
```

Untuk iterasi ini tidak dapat dilakukan terhadap interval.

>niterate("f",1,2,5)

```
[ ~1,2~,  ~0.75,1.5~,  ~0.85,1.8~,  ~0.79,1.6~,  ~0.82,1.7~,  ~0.81,1.7~ ]
```

Perhatikan, hasil iterasinya sama dengan interval awal. Alasannya adalah perhitungan dengan interval bersifat terlalu longgar. Untuk meingkatkan perhitungan pada ekspresi dapat digunakan pembagian intervalnya, menggunakan fungsi ieval().

>function s(x) := ieval("(x+2/x)/2",x,10)

Selanjutnya dapat dilakukan iterasi hingga diperoleh hasil optimal, dan intervalnya tidak semakin mengecil. Hasilnya berupa interval yang memuat akar persamaan:

$$x = \frac{1}{2}\left(x + \frac{2}{x}\right).$$

Satu-satunya solusi adalah

$$x = \sqrt{2}.$$

>iterate("s",1,2)

```
~1.41421356236,1.41421356239~
```

Fungsi "iterate()" juga dapat bekerja pada vektor. Berikut adalah contoh fungsi vektor, yang menghasilkan rata-rata aritmetika dan rata-rata geometri.

$$(a_{n+1}, b_{n+1}) = \left(\frac{a_n + b_n}{2}, \sqrt{a_n b_n}\right)$$

Iterasi ke-n disimpan pada vektor kolom x[n].

>function g(x) := [(x[1]+x[2])/2;sqrt(x[1]*x[2])]

Iterasi dengan menggunakan fungsi tersebut akan konvergen ke rata-rata aritmetika dan geometri dari nilai-nilai awal.

>iterate("g",[1;5])

```
2.60401
2.60401
```

Hasil tersebut konvergen agak cepat, seperti kita cek sebagai berikut.

>iterate("g",[1;5],4)

```
        1           3     2.61803     2.60403     2.60401
        5     2.23607     2.59002     2.60399     2.60401
```

Iterasi pada interval dapat dilakukan dan stabil, namun tidak menunjukkan bahwa limitnya pada batas-batas yang dihitung.

>iterate("g",[~1~;~5~],4)

```
Interval 2 x 5 matrix

~0.99999999999999778,1.0000000000000022~      ...
~4.9999999999999911,5.0000000000000089~      ...
```

Iterasi berikut konvergen sangat lambat.

$$x_{n+1} = \sqrt{x_n}.$$

>iterate("sqrt(x)",2,10)

```
[2,  1.41421,  1.18921,  1.09051,  1.04427,  1.0219,  1.01089,
1.00543,  1.00271,  1.00135,  1.00068]
```

Kekonvergenan iterasi tersebut dapat dipercepatdengan percepatan Steffenson:

>steffenson("sqrt(x)",2,10)

```
[1.04888,  1.00028,  1,  1]
```

Iterasi menggunakan Loop yang Ditulis Langsung

Berikut adalah beberapa contoh penggunaan loop untuk melakukan iterasi yang ditulis langsung pada baris perintah.

>x=2; repeat x=(x+2/x)/2; until x^2~=2; end; x,

```
1.41421356237
```

Penggabungan matriks menggunakan tanda "|" dapat digunakan untuk menyimpan semua hasil iterasi.

>v=[1]; for i=2 to 8; v=v|(v[i-1]*i); end; v,

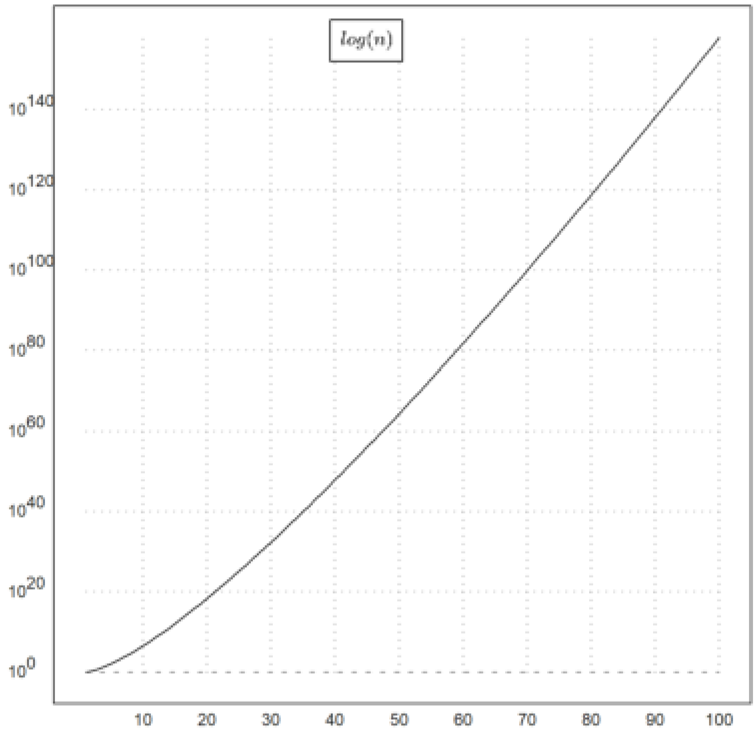
```
[1,  2,  6,  24,  120,  720,  5040,  40320]
```

hasil iterasi juga dapat disimpan pada vektor yang sudah ada.

>v=ones(1,100); for i=2 to cols(v); v[i]=v[i-1]*i; end; ...
> plot2d(v,logplot=1); textbox(latex(&log(n)),x=0.5):

>A =[0.5,0.2;0.7,0.1]; b=[2;2]; ...
> x=[1;1]; repeat xnew=A.x-b; until all(xnew~=x); x=xnew; end; ...
> x,

```
-7.09677
-7.74194
```



Gambar 382: images/KALKULUS_Diva%20Nagita(23030630024)-353.png

Iterasi di dalam Fungsi

Fungsi atau program juga dapat menggunakan iterasi dan dapat digunakan untuk melakukan iterasi. Berikut adalah beberapa contoh iterasi di dalam fungsi.

Contoh berikut adalah suatu fungsi untuk menghitung berapa lama suatu iterasi konvergen. Nilai fungsi tersebut adalah hasil akhir iterasi dan banyak iterasi sampai konvergen.

>function map hiter(f$,x0) ...

```
x=x0;
maxiter=0;
repeat
  xnew=f$(x);
  maxiter=maxiter+1;
  until xnew~=x;
  x=xnew;
end;
return maxiter;
endfunction
```

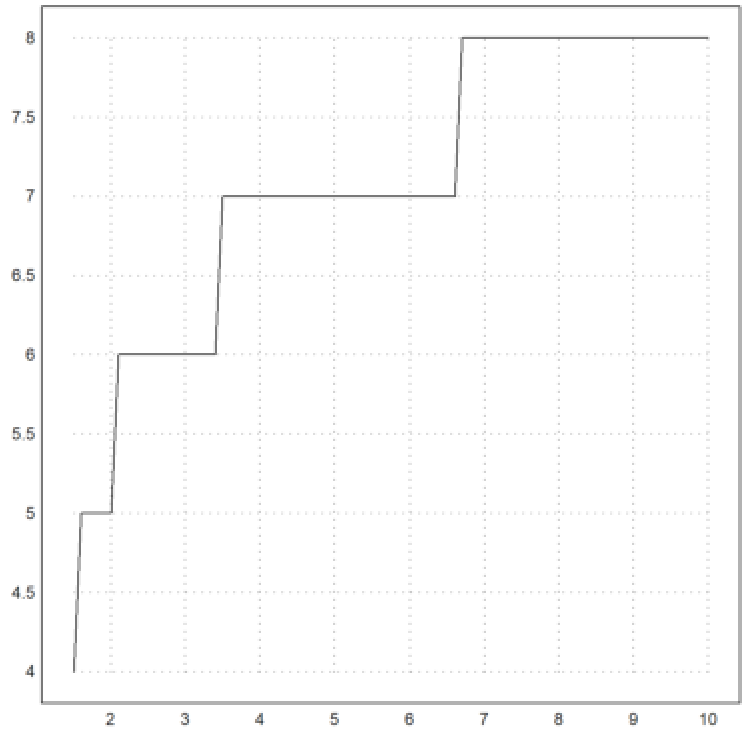
Misalnya, berikut adalah iterasi untuk mendapatkan hampiran akar kuadrat 2, cukup cepat, konvergen pada iterasi ke-5, jika dimulai dari hampiran awal 2.

>hiter("(x+2/x)/2",2)

```
5
```

Karena fungsinya didefinisikan menggunakan "map". maka nilai awalnya dapat berupa vektor.

>x=1.5:0.1:10; hasil=hiter("(x+2/x)/2",x); ...
> plot2d(x,hasil):



Gambar 383: images/KALKULUS_Diva%20Nagita(23030630024)-354.png

Dari gambar di atas terlihat bahwa kekonvergenan iterasinya semakin lambat, untuk nilai awal semakin besar, namun penambahnnya tidak kontinu. Kita dapat menemukan kapan maksimum iterasinya bertambah.

>hasil[1:10]

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

>x[nonzeros(differences(hasil))]

```
[1.5,  2,  3.4,  6.6]
```

maksimum iterasi sampai konvergen meningkat pada saat nilai awalnya 1.5, 2, 3.4, dan 6.6.

Contoh berikutnya adalah metode Newton pada polinomial kompleks berderajat 3.

>p &= x^3-1; newton &= x-p/diff(p,x); $newton

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

Selanjutnya didefinisikan fungsi untuk melakukan iterasi (aslinya 10 kali).

>function iterasi(f$,x,n=10) . . .

```
loop 1 to n; x=f$(x); end;
return x;
endfunction
```

Kita mulai dengan menentukan titik-titik grid pada bidang kompleksnya.

>r=1.5; x=linspace(-r,r,501); Z=x+I*x'; W=iterasi(newton,Z);

Berikut adalah akar-akar polinomial di atas.

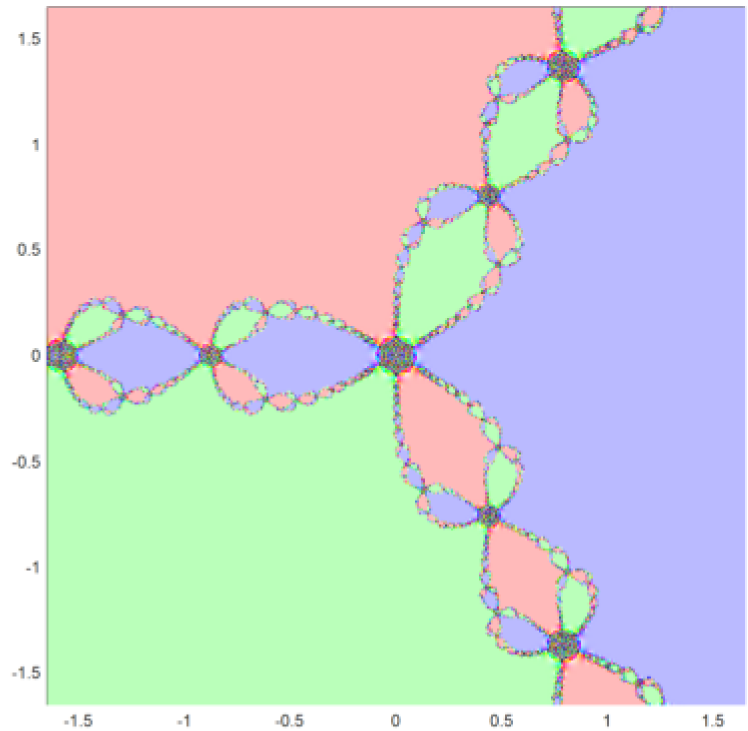
>z=&solve(p)()

```
[ -0.5+0.866025i,  -0.5-0.866025i,  1+0i  ]
```

Untuk menggambar hasil iterasinya, dihitung jarak dari hasil iterasi ke-10 ke masing-masing akar, kemudian digunakan untuk menghitung warna yang akan digambar, yang menunjukkan limit untuk masing-masing nilai awal.

Fungsi plotrgb() menggunakan jendela gambar terkini untuk menggambar warna RGB sebagai matriks.

>C=rgb(max(abs(W-z[1]),1),max(abs(W-z[2]),1),max(abs(W-z[3]),1)); . . .
> plot2d(none,-r,r,-r,r); plotrgb(C):



Gambar 384: images/KALKULUS_Diva%20Nagita(23030630024)-356.png

Iterasi Simbolik

Seperti sudah dibahas sebelumnya, untuk menghasilkan barisan ekspresi simbolik dengan Maxima dapat digunakan fungsi makelist().

>&powerdisp:true // untuk menampilkan deret pangkat mulai dari suku berpangkat terkecil

```
true
```

>deret &= makelist(taylor(exp(x),x,0,k),k,1,3); $deret // barisan deret Taylor untuk e^x

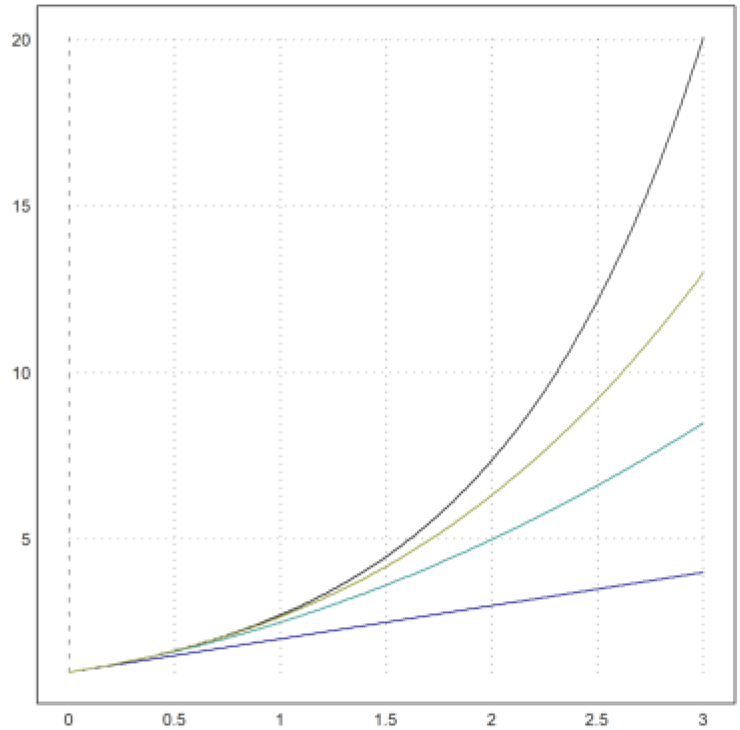
$$\left[1+x,1+x+\frac{x^2}{2},1+x+\frac{x^2}{2}+\frac{x^3}{6}\right]$$

Untuk mengubah barisan deret tersebut menjadi vektor string di EMT digunakan fungsi mxm2str(). Selanjutnya, vektor string/ekspresi hasilnya dapat digambar seperti menggambar vektor eskpresi pada EMT.

>plot2d("exp(x)",0,3); // plot fungsi aslinya, e^x

>plot2d(mxm2str("deret"),>add,color=4:6): // plot ketiga deret taylor hampiran fungsi tersebut

Selain cara di atas dapat juga dengan cara menggunakan indeks pada vektor/list yang dihasilkan.



Gambar 385: images/KALKULUS_Diva%20Nagita(23030630024)-358.png

>$deret[3]

$$1 + x + \frac{x^2}{2} + \frac{x^3}{6}$$

>plot2d(["exp(x)",&deret[1],&deret[2],&deret[3]],0,3,color=1:4):

>$sum(sin(k*x)/k,k,1,5)

$$\sin x + \frac{\sin(2\,x)}{2} + \frac{\sin(3\,x)}{3} + \frac{\sin(4\,x)}{4} + \frac{\sin(5\,x)}{5}$$

Berikut adalah cara menggambar kurva

$$y = \sin(x) + \frac{\sin 3x}{3} + \frac{\sin 5x}{5} + \ldots.$$

>plot2d(&sum(sin((2*k+1)*x)/(2*k+1),k,0,20),0,2pi):

Hal serupa juga dapat dilakukan dengan menggunakan matriks, misalkan kita akan menggambar kurva

$$y = \sum_{k=1}^{100} \frac{\sin(kx)}{k}, \quad 0 \le x \le 2\pi.$$

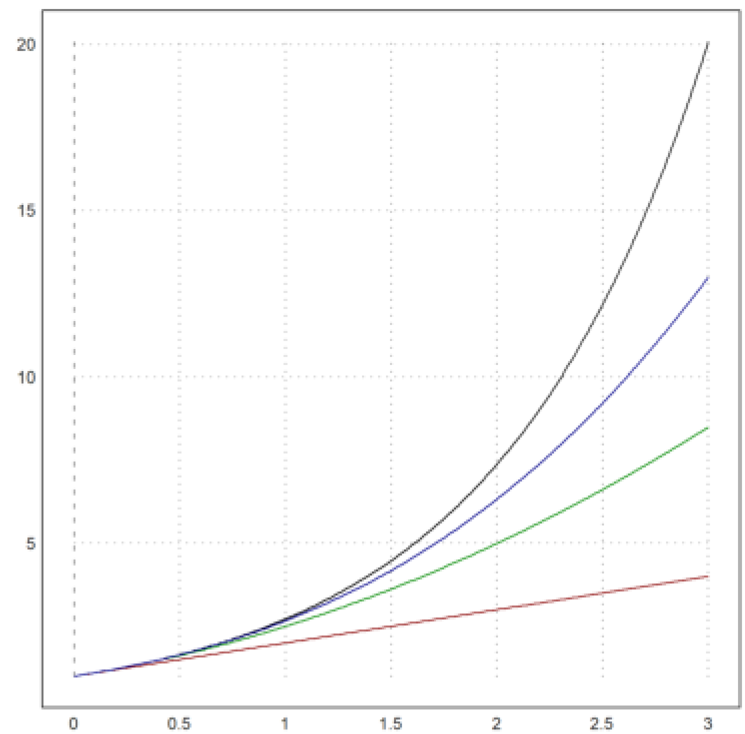
>x=linspace(0,2pi,1000); k=1:100; y=sum(sin(k*x')/k)'; plot2d(x,y):

Tabel Fungsi

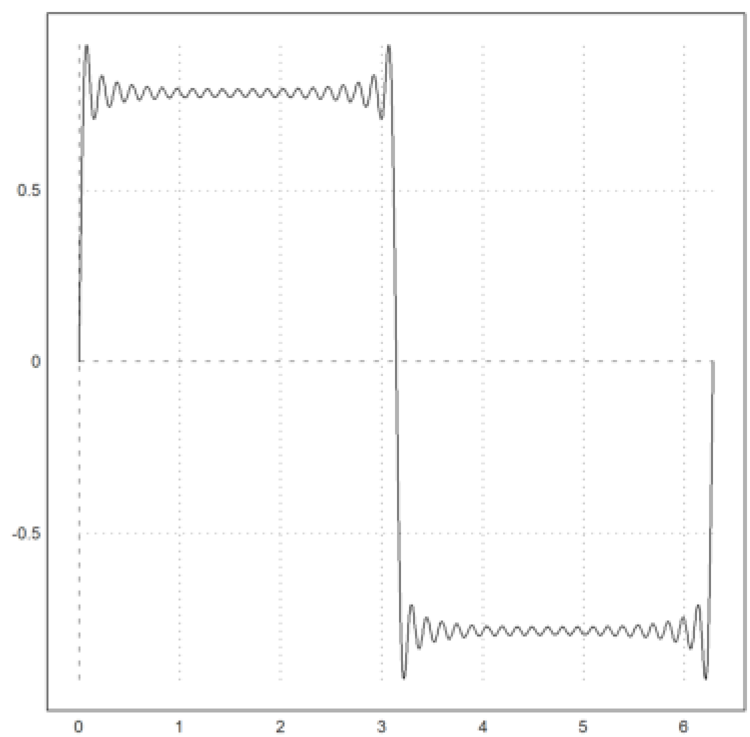
Terdapat cara menarik untuk menghasilkan barisan dengan ekspresi Maxima. Perintah mxmtable() berguna untuk menampilkan dan menggambar barisan dan menghasilkan barisan sebagai vektor kolom.

Sebagai contoh berikut adalah barisan turunan ke-n x^x di x=1.

>mxmtable("diffat(x^x,x=1,n)","n",1,8,frac=1);

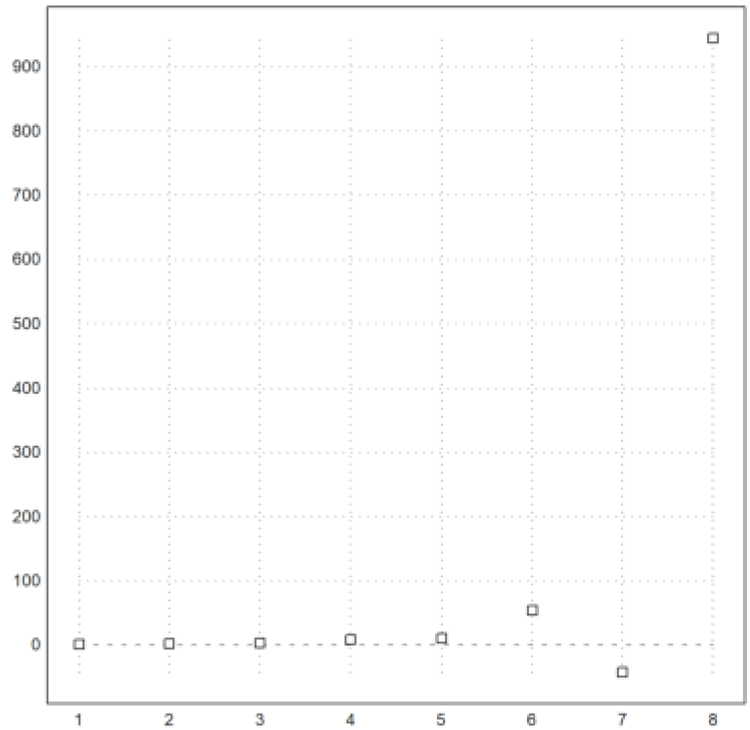


Gambar 386: images/KALKULUS_Diva%20Nagita(23030630024)-360.png



Gambar 387: images/KALKULUS_Diva%20Nagita(23030630024)-363.png

```
  1
  2
  3
  8
 10
 54
-42
944
```



Gambar 388: images/KALKULUS_Diva%20Nagita(23030630024)-366.png

>$'sum(k, k, 1, n) = factor(ev(sum(k, k, 1, n),simpsum=true)) // simpsum:menghitung deret secara simbolik

$$\sum_{k=1}^{n} k = \frac{n\,(1+n)}{2}$$

>$'sum(1/(3^k+k), k, 0, inf) = factor(ev(sum(1/(3^k+k), k, 0, inf),simpsum=true))

$$\sum_{k=0}^{\infty} \frac{1}{k+3^k} = \sum_{k=0}^{\infty} \frac{1}{k+3^k}$$

Di sini masih gagal, hasilnya tidak dihitung.

>$'sum(1/x^2, x, 1, inf)= ev(sum(1/x^2, x, 1, inf),simpsum=true) // ev: menghitung nilai ekspresi

$$\sum_{x=1}^{\infty} \frac{1}{x^2} = \frac{\pi^2}{6}$$

>$'sum((-1)^(k-1)/k, k, 1, inf) = factor(ev(sum((-1)^(x-1)/x, x, 1, inf),simpsum=true))

$$\sum_{k=1}^{\infty} \frac{(-1)^{-1+k}}{k} = -\sum_{x=1}^{\infty} \frac{(-1)^x}{x}$$

Di sini masih gagal, hasilnya tidak dihitung.

>$’sum((-1)^k/(2*k-1), k, 1, inf) = factor(ev(sum((-1)^k/(2*k-1), k, 1, inf),simpsum=true))

$$\sum_{k=1}^{\infty}{\frac{\left(-1\right)^{k}}{-1+2\,k}}=\sum_{k=1}^{\infty}{\frac{\left(-1\right)^{k}}{-1+2\,k}}$$

>$ev(sum(1/n!, n, 0, inf),simpsum=true)

$$\sum_{n=0}^{\infty}{\frac{1}{n!}}$$

Di sini masih gagal, hasilnya tidak dihitung, harusnya hasilnya e.

>&assume(abs(x)<1); $’sum(a*x^k, k, 0, inf)=ev(sum(a*x^k, k, 0, inf),simpsum=true), &forget(abs(x)<1);

$$a\,\sum_{k=0}^{\infty}{x^{k}}=\frac{a}{1-x}$$

Deret geometri tak hingga, dengan asumsi rasional antara -1 dan 1.

>$’sum(x$^{\text{k/k!,k,0,inf)=ev(sum(x}}$k/k!,k,0,inf),simpsum=true)

$$\sum_{k=0}^{\infty}{\frac{x^{k}}{k!}}=\sum_{k=0}^{\infty}{\frac{x^{k}}{k!}}$$

>$limit(sum(x^k/k!,k,0,n),n,inf)

$$\lim_{n\rightarrow\infty}{\sum_{k=0}^{n}{\frac{x^{k}}{k!}}}$$

>function d(n) &= sum(1/(k^2-k),k,2,n); $’d(n)=d(n)

$$d\left(n\right)=\sum_{k=2}^{n}{\frac{1}{-k+k^{2}}}$$

>$d(10)=ev(d(10),simpsum=true)

$$\sum_{k=2}^{10}{\frac{1}{-k+k^{2}}}=\frac{9}{10}$$

>$d(100)=ev(d(100),simpsum=true)

$$\sum_{k=2}^{100}{\frac{1}{-k+k^{2}}}=\frac{99}{100}$$

Deret Taylor

Deret Taylor suatu fungsi f yang diferensiabel sampai tak hingga di sekitar x=a adalah:

$$f(x)=\sum_{k=0}^{\infty}\frac{(x-a)^{k}f^{(k)}(a)}{k!}.$$

>$’e^x =taylor(exp(x),x,0,10) // deret Taylor e^x di sekitar x=0, sampai suku ke-11

$$e^{x}=1+x+\frac{x^{2}}{2}+\frac{x^{3}}{6}+\frac{x^{4}}{24}+\frac{x^{5}}{120}+\frac{x^{6}}{720}+\frac{x^{7}}{5040}+\frac{x^{8}}{40320}+\frac{x^{9}}{362880}+\frac{x^{10}}{3628800}$$

>$’log(x)=taylor(log(x),x,1,10)// deret log(x) di sekitar x=1

$$\log x=-1-\frac{\left(-1+x\right)^{2}}{2}+\frac{\left(-1+x\right)^{3}}{3}-\frac{\left(-1+x\right)^{4}}{4}+\frac{\left(-1+x\right)^{5}}{5}-\frac{\left(-1+x\right)^{6}}{6}+\frac{\left(-1+x\right)^{7}}{7}-\frac{\left(-1+x\right)^{8}}{8}+\frac{\left(-1+x\right)^{9}}{9}-\frac{\left(-1+x\right)^{10}}{10}+x$$

—

Kita juga dapat menggunakan fungsi “sequence()” untuk barisan yang kompleks. Fungsi ini menghitung nilai-nilai x[n] dari semua nilai sebelumnya, x[1],. . . ,x[n-1] yang diketahui.

Berikut adalah contoh barisan Fibonacci.

$$x_n = x_{n-1} + x_{n-2}, \quad x_1 = 2, \quad x_2 = 3$$

>sequence(“x[n-1]+x[n-2]”,[2,3],10)

```
[2,  3,  5,  8,  13,  21,  34,  55,  89,  144]
```

Kemudian kita dapat menggunakan fungsi makelist() untuk menghasilkan barisan ekspresi simbolik dengan maxima.

>&powerdisp:true

```
true
```

perintah tersebut untuk menampilkan deret pangkat mulai dari suku berpangkat terkecil

Permukaan dalam Rˆ3 terdapat dua macam yakni permukaan linear dan

kuadratik. Setiap permukaan linear berupa bidang datar, sedangkan

permukaan kuadratik berupa bidang lengkung yang kelengkungannya

bergantung atas bentuk persamaannya.

Untuk membuat grafik fungsi tiga dimensi, maka kita dapat menggunakan

perintah “plot3d()”

Grafik Fungsi Persamaan Linear Dalam Dimensi Tiga

Bentuk umum persamaan permukaan linear adalah

$$Ax + By + Cz + D = 0$$

Contoh grafik persamaan linear di ruang tiga dimensi:

>deret &= makelist(taylor(exp(x),x,0,k),k,2,4); $deret

$$\left[1 + x + \frac{x^2}{2}, 1 + x + \frac{x^2}{2} + \frac{x^3}{6}, 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24}\right]$$

FUNGSI MULTIVARIABEL Fungsi multivariabel adalah pemetaan matematis

yang menghubungkan beberapa variabel independen dengan satu variabel dependen. Fungsi ini umumnya dinotasikan sebagai

$f(x, y)$ atau $f(x_1, x_2, ..., x_n)$

dimana x, y atau $x_1, x_2, ..., x_n$ adalah variabel independen

dan f adalah variabel dependen.

Fungsi multivariabel dapat diwakili dalam bentuk peta atau grafik tiga

dimensi.

Contoh fungsi multivariabel :

$$z = x^2 + y^2$$

Dimana variabel bebasnya yaitu x dan y.

Grafik Fungsi Multivariabel Pada fungsi multivariabel, grafik fungsinya merupakan grafik tiga dimensi. Ruang dimensi tiga dilambangkan dengan

$$R^3$$

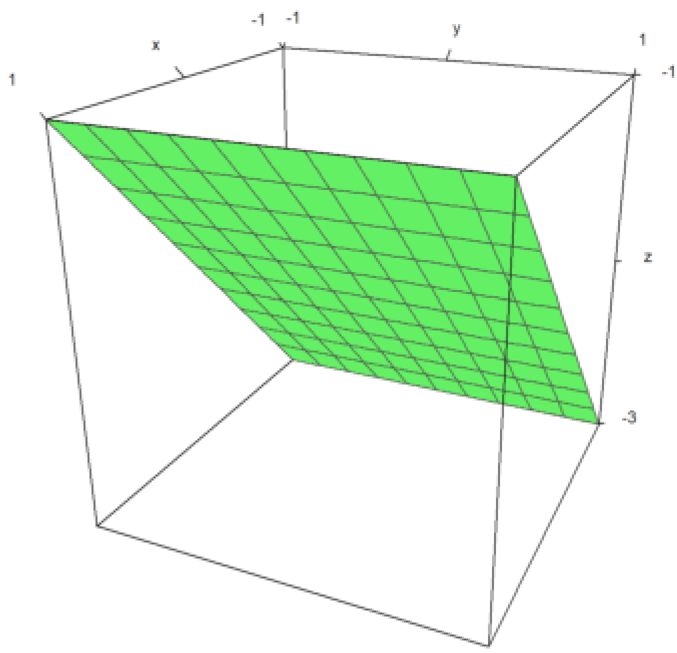
Permukaan dalam R^3 terdapat dua macam yakni permukaan linear dan
kuadratik. Setiap permukaan linear berupa bidang datar, sedangkan
permukaan kuadratik berupa bidang lengkung yang kelengkungannya
bergantung atas bentuk persamaannya.
Untuk membuat grafik fungsi tiga dimensi, maka kita dapat menggunakan
perintah "plot3d()"

Grafik Fungsi Persamaan Linear Dalam Dimensi Tiga

Bentuk umum persamaan permukaan linear adalah

$$Ax + By + Cz + D = 0$$

Contoh grafik persamaan linear di ruang tiga dimensi:
>plot3d("x-2"):

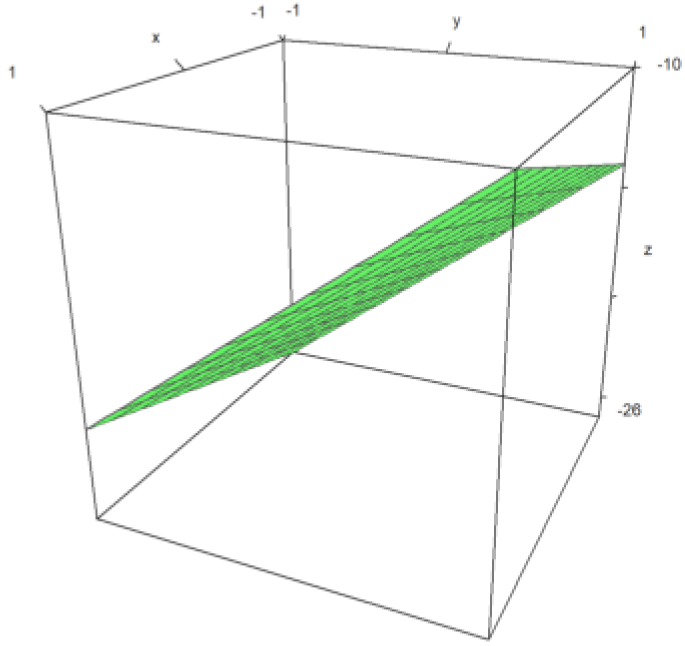


Gambar 389: images/KALKULUS_Diva%20Nagita(23030630024)-391.png

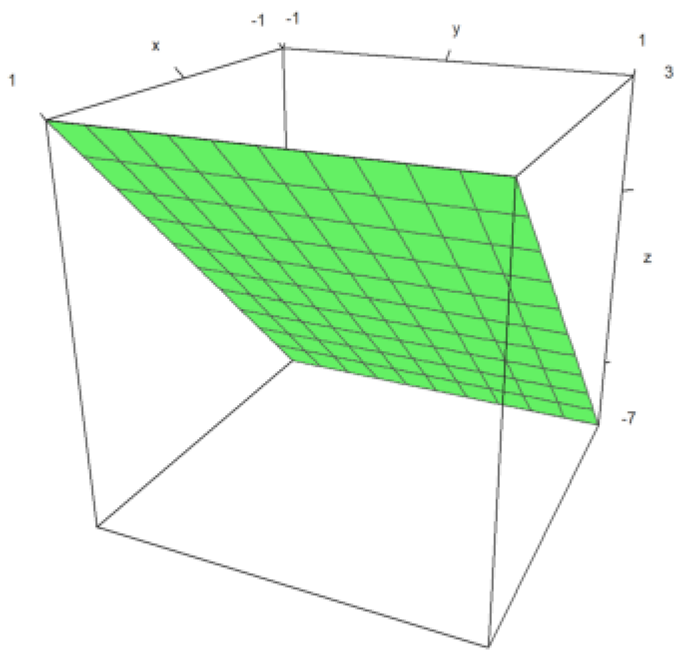
>plot3d("2*x+6*y-18"):
>plot3d("5*x-2"):
>plot3d("2*x+8*y+4*z-18",implicit=3,r=5):

Grafik Fungsi Kuadratik di Ruang Dimensi Tiga

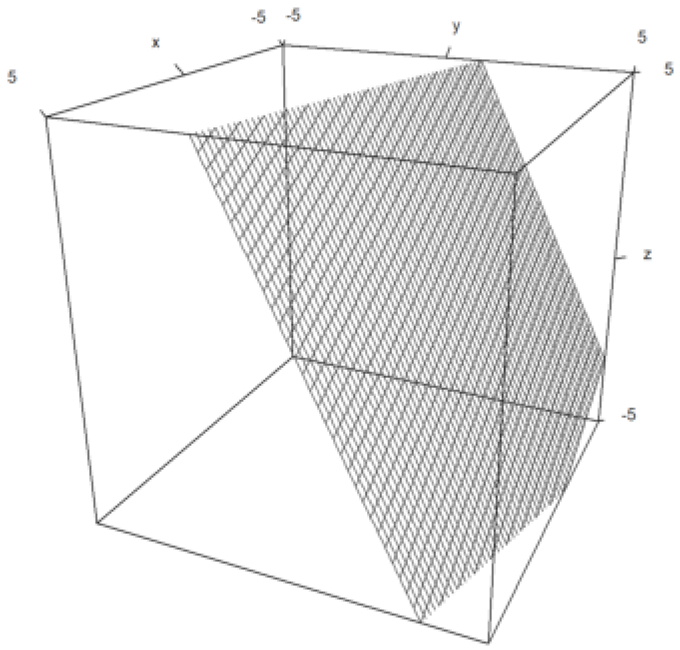
Persamaan kuadratik mempunyai rumus umum :



Gambar 390: images/KALKULUS_Diva%20Nagita(23030630024)-392.png



Gambar 391: images/KALKULUS_Diva%20Nagita(23030630024)-393.png



Gambar 392: images/KALKULUS_Diva%20Nagita(23030630024)-394.png

$$Ax^2 + By^2 + Cz^2 + Dxy + Exz + Fyz + Gx + Hy + Iz + J = 0$$

Permukaan-permukaan kuadratik dapat berupa permukaan bola, ellipsoida, paraboloida, tabung ellips, tabung lingkaran, atau tabung parabola.
Contoh grafik fungsi kuadratik di ruang dimensi tiga:
>plot3d("2*x^{2+4*y}2"):
>function f(x,y) := exp(-(2*x^{2+4*y}2))
>plot3d("f", r=5):
>plot3d("9*x^{2+4*y}2+9*z^2-36",implicit=2,r=3):
Gambar diatas merupakan ellipsoid yang persamaannya biasa dinyatakan dalam

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1$$

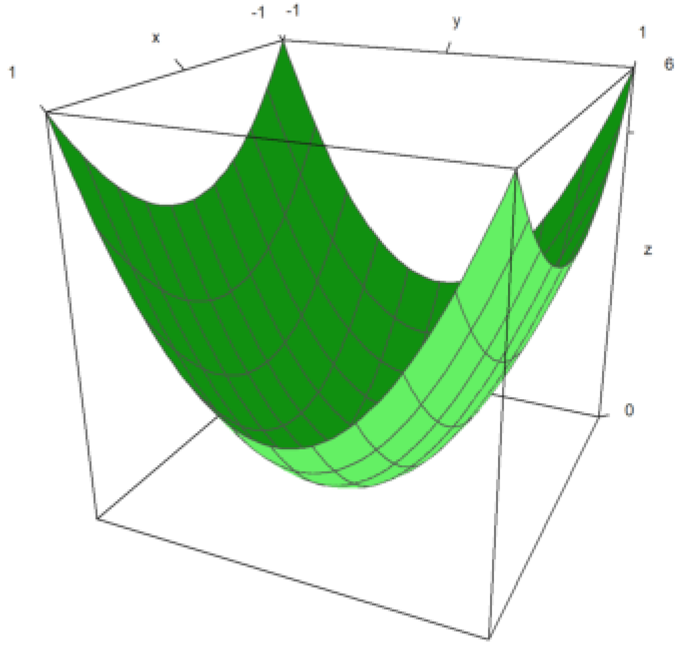
Sedangkan grafik diatas merupakan ellipsoid dengan persamaan :

$$\frac{x^2}{4} + \frac{y^2}{9} + \frac{z^2}{4} = 1$$

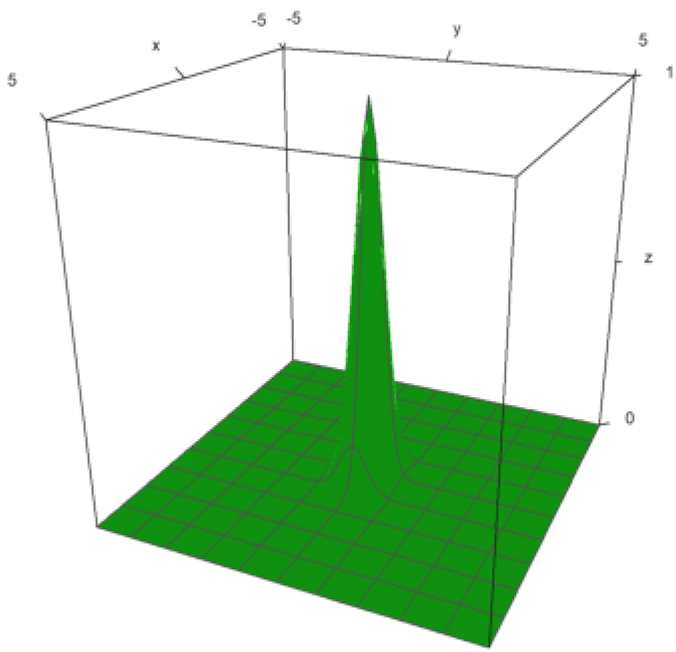
Namun, untuk memudahkan dalam memplotnya kita hilangkan bentuk pecahannya dengan mengurangkan kedua ruas dengan satu lalu mengalikannya dengan 36 sehingga diperoleh

$$9x^2 + 4y^2 + 9z^2 - 36 = 0$$

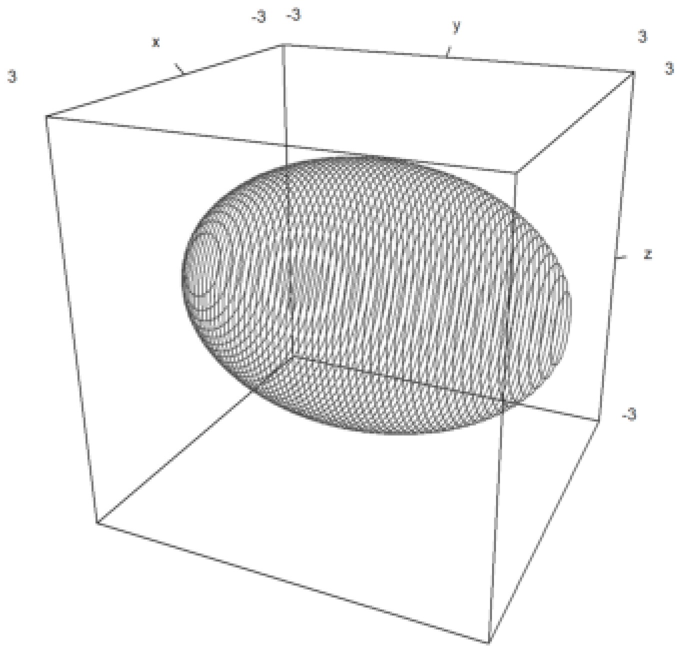
>plot3d("y^{2-x}2",r=2):
Gambar diatas merupakan paraboloida hiperbolik



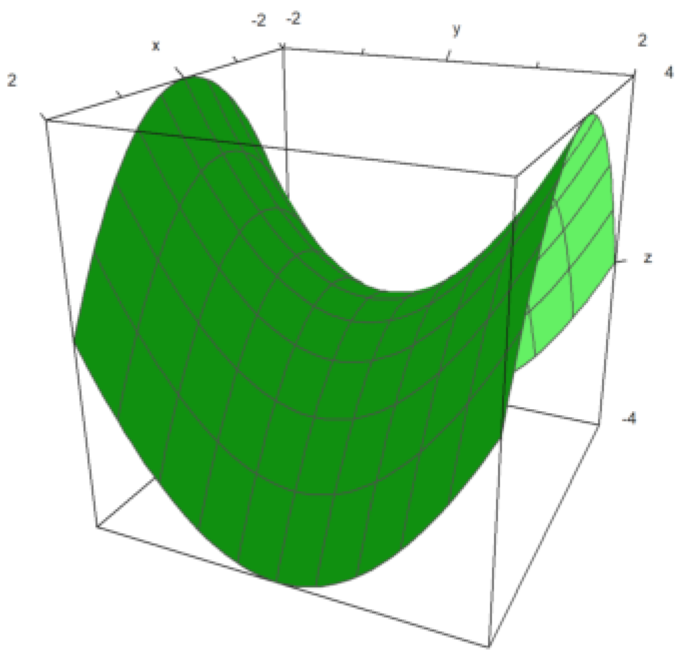
Gambar 393: images/KALKULUS_Diva%20Nagita(23030630024)-396.png



Gambar 394: images/KALKULUS_Diva%20Nagita(23030630024)-397.png

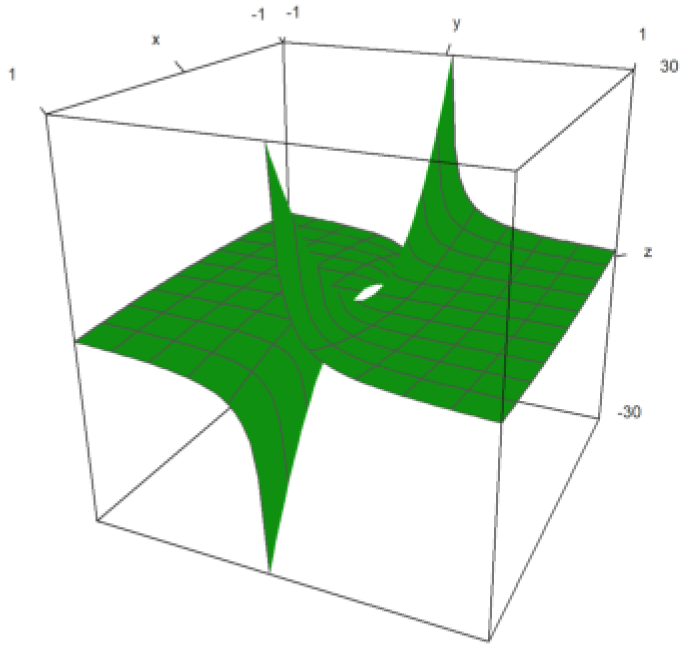


Gambar 395: images/KALKULUS_Diva%20Nagita(23030630024)-398.png



Gambar 396: images/KALKULUS_Diva%20Nagita(23030630024)-402.png

>plot3d("x^2/y"):



Gambar 397: images/KALKULUS_Diva%20Nagita(23030630024)-403.png

Menggambar Kurva Perpotongan dari Dua Persamaan

Di EMT kita juga dapat menggabungkan dua kurva pada satu bidang untuk menggambarkan perpotongan. Untuk masalah ini kita gunakan fungsi

$$> add$$

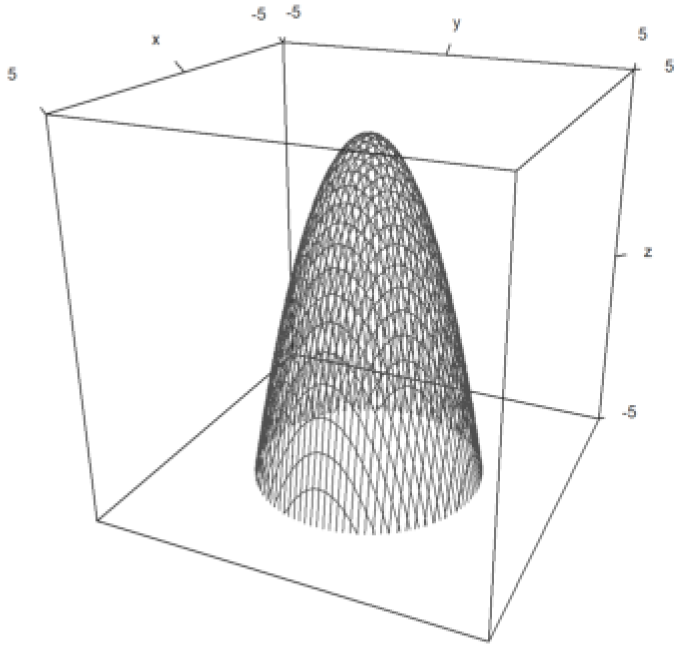
dalam prosesnya.
Contoh :

$$x^2 + y^2 + z - 4 = 0$$
dengan
$$x^2 + y^2 = 1$$

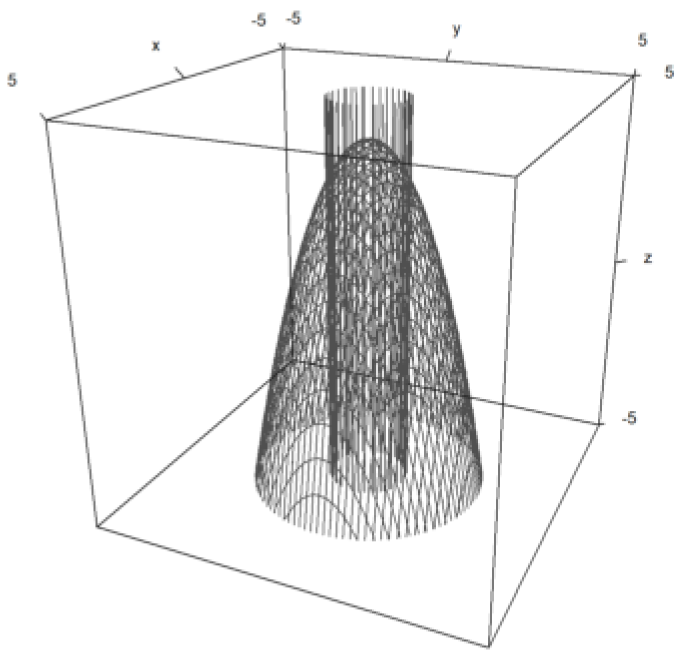
>plot3d("x^{2+y}2+z-4",r=5, implicit=3):
>plot3d("x^{2+y}2-1",implicit=3, r=5, >add):
Dari persamaan diatas, didapatkan perpotongan antara bidang datar dan bidang lengkung.

3.

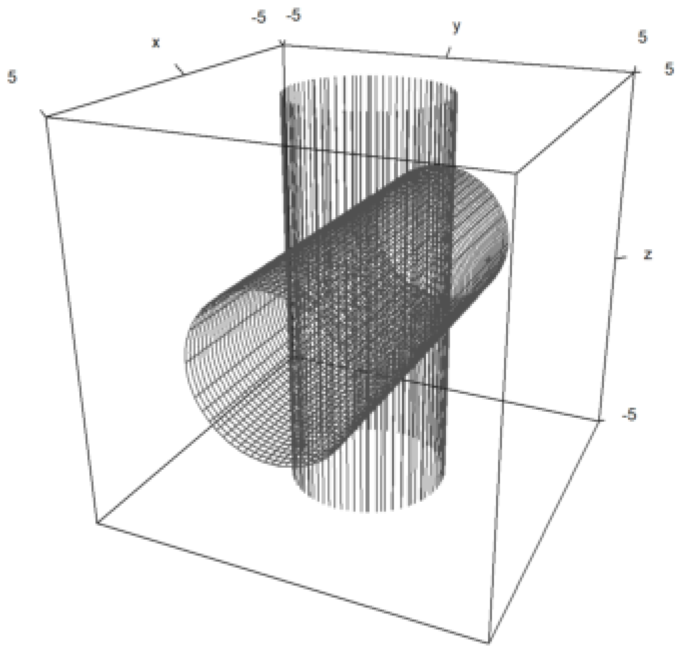
$$x^2 + y^2 = 4$$
dengan
$$y^2 + z^2 = 4$$



Gambar 398: images/KALKULUS_Diva%20Nagita(23030630024)-408.png



Gambar 399: images/KALKULUS_Diva%20Nagita(23030630024)-409.png



Gambar 400: images/KALKULUS_Diva%20Nagita(23030630024)-413.png

>plot3d("x^2+y^2-4",r=5,implicit=3); plot3d("y^2+z^2-4",r=5,implicit=3,>add):
Didapatkan perpotongan antara dua tabung lingkaran.

Turunan Fungsi Multivariabel ** Turunan Fungsi Dua Variabel

Turunan parsial, yaitu turunan fungsi terhadap satu variabel bebas sementara variabel bebas lainnya dianggap tetap atau konstan.

- Turunan Parsial terhadap f terhadap x di (x0,y0)

$$f_x(x_0, y_0) = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x, y_0) - f(x_0, y_0)}{\Delta x}$$

* Turunan Parsial f terhadap y di (x0,y0) * f_y(x_0,y_0)= $\lim$ _{$\Delta y \to 0$}$\frac{f(x_0,y_0+\Delta y)-f(x_0,y_0)}{\Delta y}$

$$f_y(x_0, y_0) = \lim_{\Delta y \to 0} \frac{f(x_0, y_0 + \Delta y) - f(x_0, y_0)}{\Delta y}$$

Contoh soal untuk turunan parsial :

1.

$$f(x, y) = 2xy - (1 - x^2)$$

>z &= 2*x*y-(1-x^2)

```
                 2
      - 1 + x   + 2 x y
```

>$showev('limit(((2*(x+h)*y)-(1-(x+h)^2))/h,h,0))

$$\lim_{h\to 0}\frac{-1+(h+x)^2+2\ (h+x)\ y}{h}=\lim_{h\to 0}\frac{-1+(h+x)^2+2\ (h+x)\ y}{h}$$

Perhitungan akan dilakukan menggunakan diff

>&diff(z,x) // z akan diturunkan terhadap x

```
2 x + 2 y
```

Karena pada fungsi z terdapat 2xy yang merupakan perkalian, untuk menghitung turunannya kita gunakan u'v+uv', sehingga turunan dari 2xy:

$$2.1.y+2.x.0=2y$$

Kemudian turunan dari variabel berpangkat yaitu:

u$^{n=n.u}${n-1}.u'

Jadi turunan dari x^2:

$$x^2=2.x^{2-1}.1=2x$$

Turunan dari konstanta adalah 0

sehingga, turunan dari fungsi

$$f(x,y)=2xy+x^2-1$$

terhadap x adalah

$$f_x(x,y)=2y+2x$$

>&diff(z,y) // z akan diturunkan terhadap y

```
2 x
```

Seperti pada turunan terhadap x, kita gunakan langkah yang sama namun kita anggap x konstan.

$$f(x,y)=2xy+x^2-1$$
$$f_y(x,y)=2.x.1+0-0$$
$$f_y(x,y)=2x$$

Jadi, turunan terhadap y dari f(x,y) adalah 2x.

Sehingga, turunan parsial dari

$$f(x,y)=2xy+x^2-1 \text{ adalah}$$
$$f_x(x,y)=2x+2y$$
$$f_y(x,y)=2x$$

2. Hitunglah turunan terhadap x dan terhadap y dari

$$f(x,y)=x^2cos(xy)$$

>expr&=x^2*cos(x*y); // definisikan fungsi f(x,y)

>&diff(expr,x)

```
                   2
2 x cos(x y) - x  y sin(x y)
```

x disini untuk menandakan bahwa fungsi diturunkan terhadap x dan y dianggap konstan.
>&diff(expr,y)

```
       3
    - x  sin(x y)
```

y menandakan bahwa fungsi diturunkan terhadap y dan x dianggap konstan.
Jadi, turunan dari fungsi

$$x^2 cos(xy)$$

$$\text{terhadap x : } 2x.cos(xy) - x^2 y.sin(xy)$$

$$\text{terhadap y : } -x^3 sin(xy)$$

Turunan Orde Tinggi
Turunan parsial kedua meliputi :

$$f_{xx} = \frac{\partial^2 f}{\partial x^2}$$

Fungsi f diturunkan terhadap x kemudian turunan pertama diturunkan lagi terhadap x.

$$f_{xy} = \frac{\partial^2 f}{\partial y \partial x}$$

Fungsi f diturunkan terhadap x kemudian turunan pertamanya diturunkan lagi terhadap y.

$$f_{yy} = \frac{\partial^2 f}{\partial y^2}$$

Fungsi f diturunkan terhadap y kemudian turunan pertama diturunkan lagi terhadap y.

$$f_{yx} = \frac{\partial^2 f}{\partial x \partial y}$$

Fungsi f diturunkan terhadap y kemudian turunan pertama diturunkan lagi terhadap x.
Turunan parsial tingkat tiga dan lebih tinggi didefinisikan dengan cara yang sama dan cara penulisannya pun serupa. Jadi, jika suatu fungsi dua variabel x dan y, turunan parsial ketiga f yang diperoleh dengan mendiferensialkan f secara parsial, pertamakali terhadap x dan kemudian terhadap y, akan ditunjukkan oleh

$$f_{xyy} = \frac{\partial^3 f}{\partial y^2 \partial x}$$

Seluruhnya, terdapat delapan turunan parsial ketiga.
Contoh :

Carilah

$$f_{xx} \text{dan} f_{yy}$$

dari fungsi berikut

$$f(x,y) = xe^y - sin(\frac{x}{y}) + x^3y^2$$

Latihan Soal

>function a(x,y) ...

```
return x^2+y^2-24
endfunction
```

>function q(x,y) ...

```
return y^2/(x^2/3)
endfunction
```

>q(4,2), q(2,3), q(4,3)

```
0.75
6.75
1.6875
```

>a(2,1), a(5,4), a(2,4)

```
-19
17
-4
```

unicodehyperref hyphensurl [ ]book xcolor amsmath,amssymb iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype [protrusion]basicmath parskip graphicx bookmark xurl same hidelinks, pdfcreator=LaTeX via pandoc

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.

Luas, Lingkaran Luar, Lingkaran Dalam Segitiga

Untuk menggambar objek-objek geometri, langkah pertama adalah menentukan rentang sumbu-sumbu koordinat. Semua objek geometri akan digambar pada satu bidang koordinat, sampai didefinisikan bidang koordinat yang baru.

>setPlotRange(-0.5,2.5,-0.5,2.5): // mendefinisikan bidang koordinat baru

Sekarang tetapkan tiga titik dan plotkan.

>A=[1,0]; plotPoint(A,"A"): // definisi dan gambar tiga titik

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

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

Kemudian tiga segmen.

>plotSegment(A,B,"c"): // c=AB

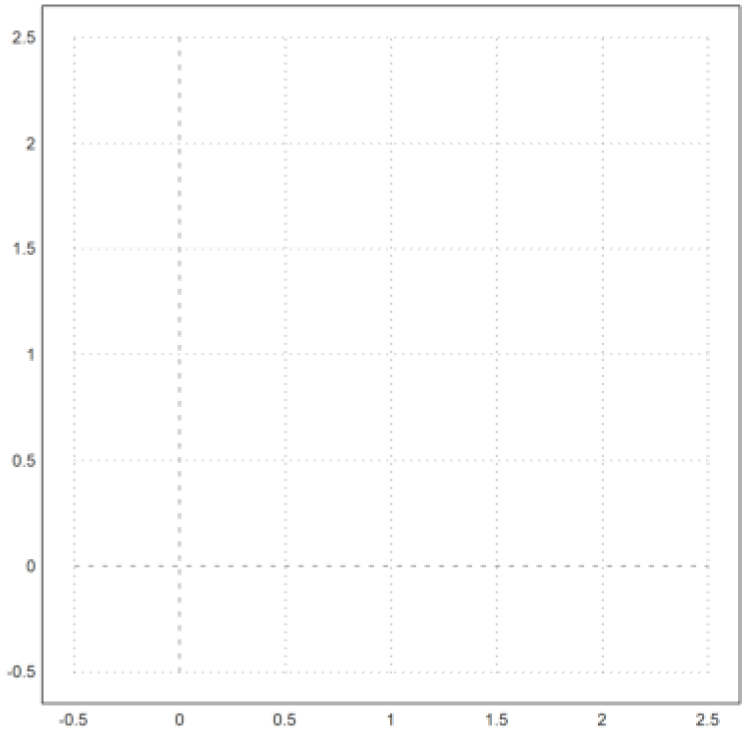
>plotSegment(B,C,"a"): // a=BC

>plotSegment(A,C,"b"): // b=AC

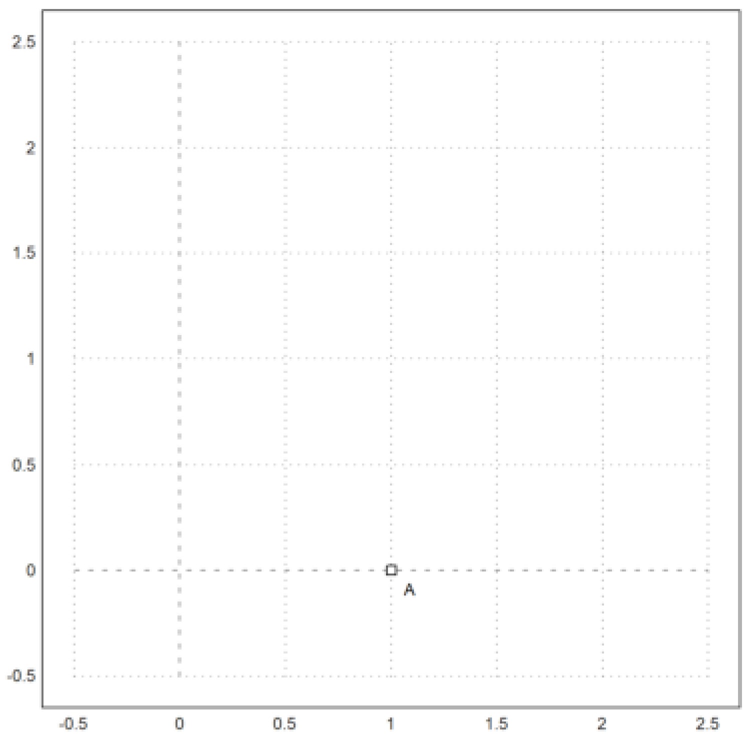
Fungsi geometri mencakup fungsi untuk membuat garis dan lingkaran. Format untuk garis adalah [a,b,c], yang merepresentasikan garis dengan persamaan ax+by=c.

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

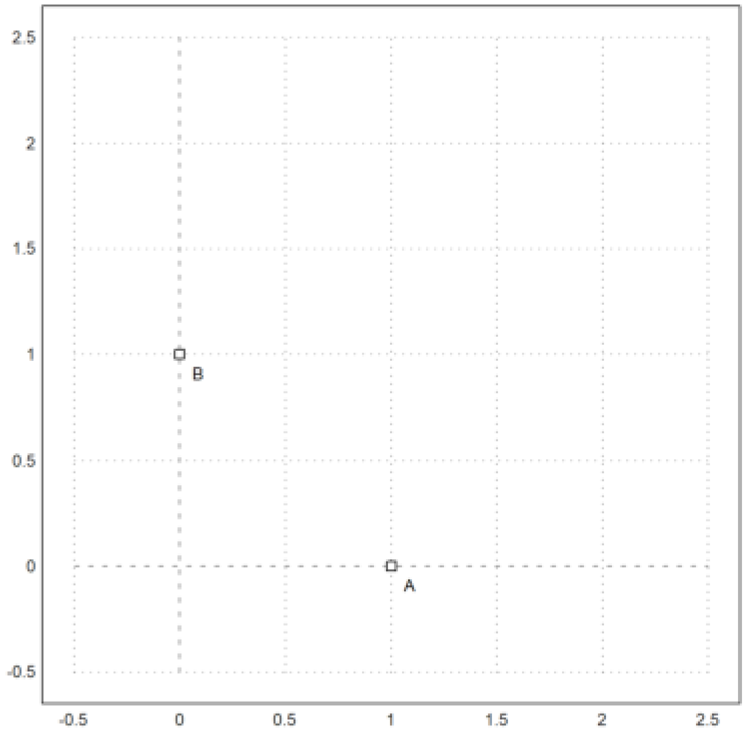
```
[-1,  2,  2]
```



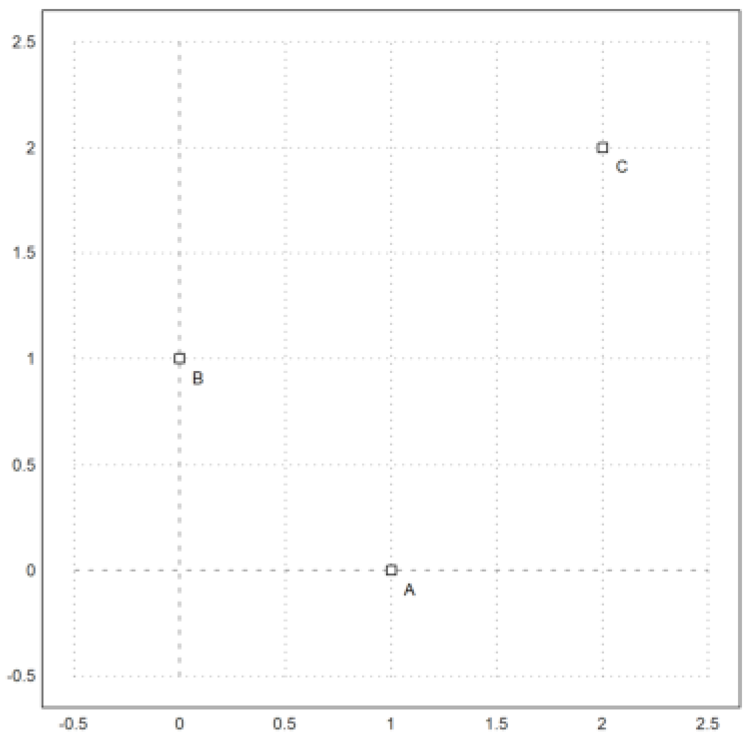
Gambar 401: images/GEOMETRI_Diva%20Nagita_(23030630024)-001.png



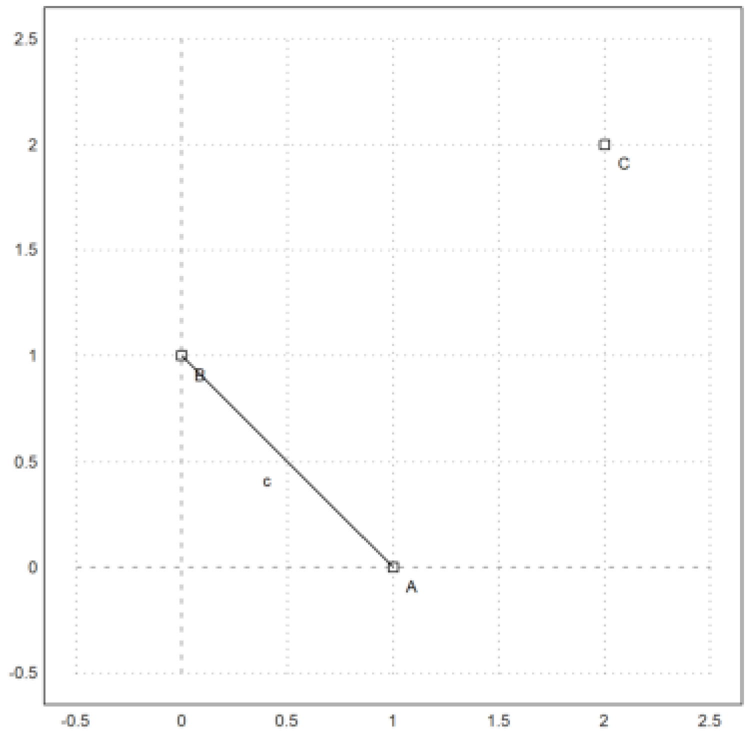
Gambar 402: images/GEOMETRI_Diva%20Nagita_(23030630024)-002.png



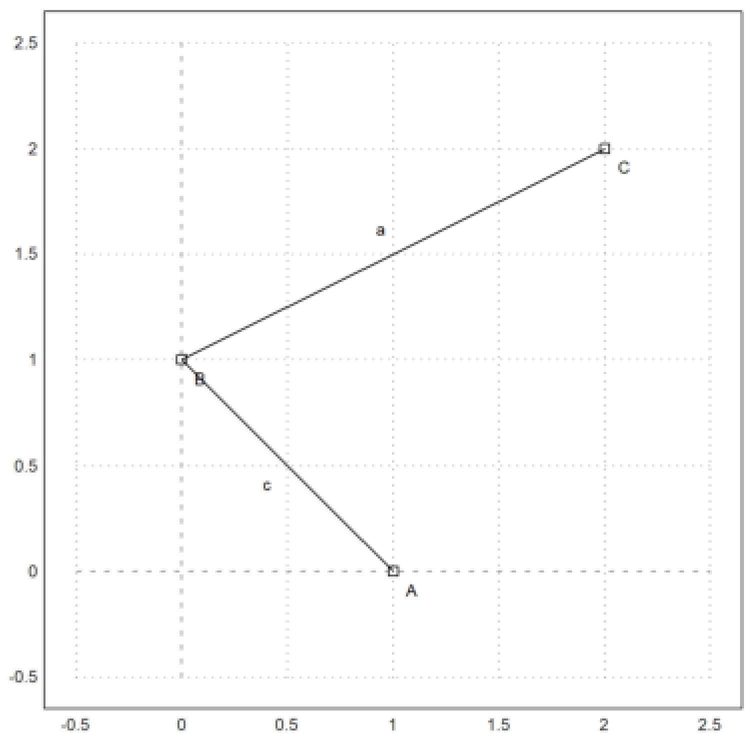
Gambar 403: images/GEOMETRI_Diva%20Nagita_(23030630024)-003.png



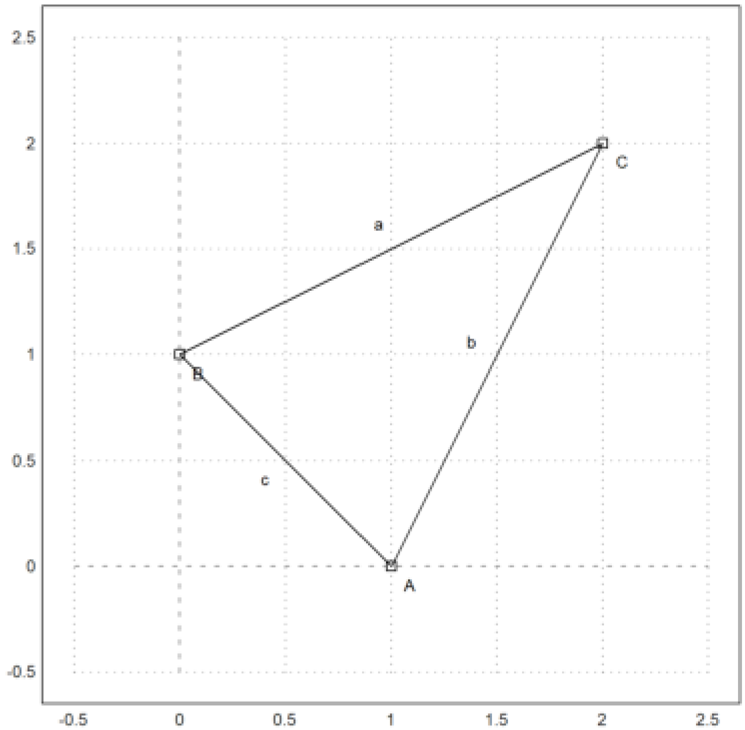
Gambar 404: images/GEOMETRI_Diva%20Nagita_(23030630024)-004.png



Gambar 405: images/GEOMETRI_Diva%20Nagita_(23030630024)-005.png



Gambar 406: images/GEOMETRI_Diva%20Nagita_(23030630024)-006.png



Gambar 407: images/GEOMETRI_Diva%20Nagita_(23030630024)-007.png

Hitung garis tegak lurus yang melalui A pada BC.
>h=perpendicular(A,lineThrough(B,C)): // garis h tegak lurus BC melalui A
dan persimpangannya dengan BC.
>D=lineIntersection(h,lineThrough(B,C)): // D adalah titik potong h dan BC
plotkan.
>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)

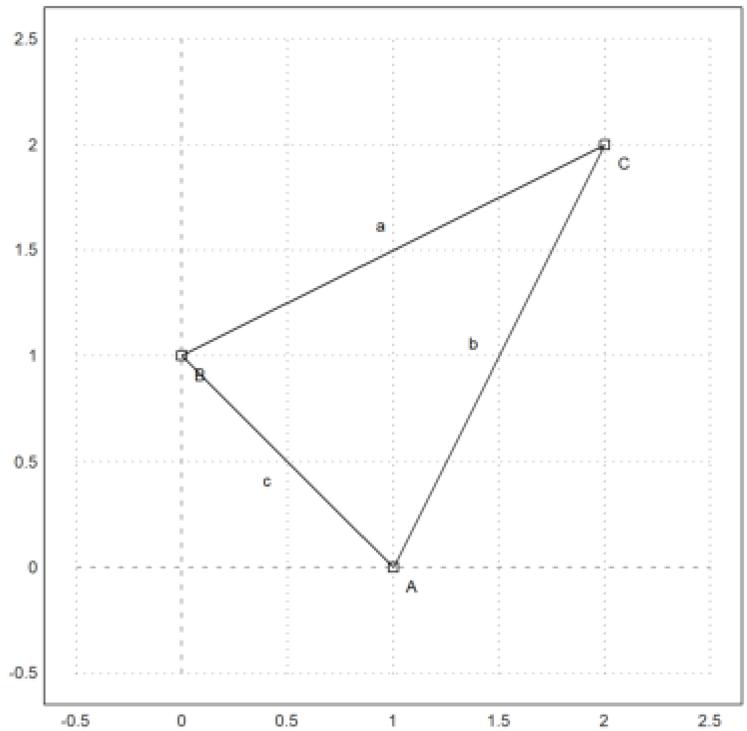
```
1.5
```

Bandingkan dengan rumus determinan.
>areaTriangle(A,B,C): // hitung luas segitiga langusng dengan fungsi
Cara lain menghitung luas segitigas ABC:
>distance(A,D)*distance(B,C)/2

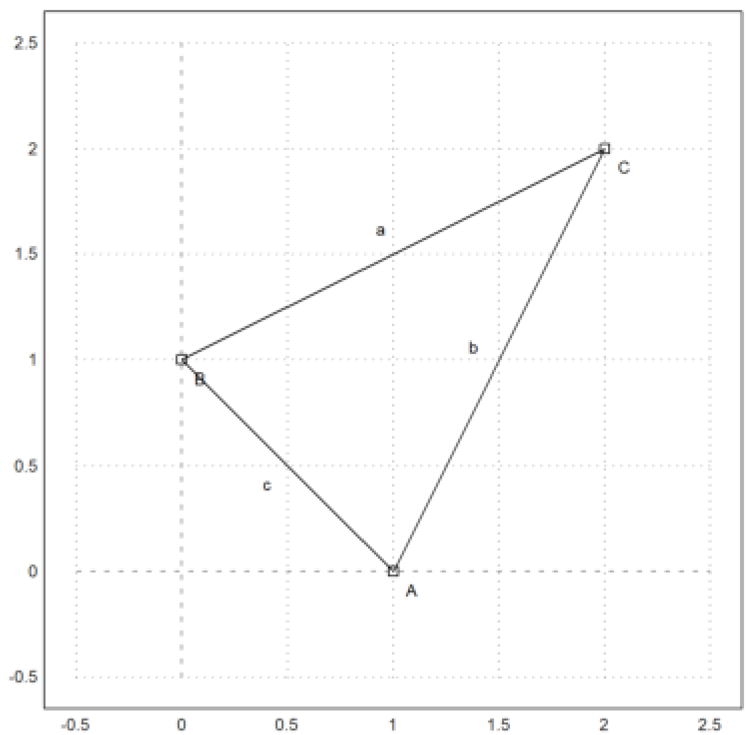
```
1.5
```

Sudut pada C.
>degprint(computeAngle(B,C,A))// menentukan besar sudut

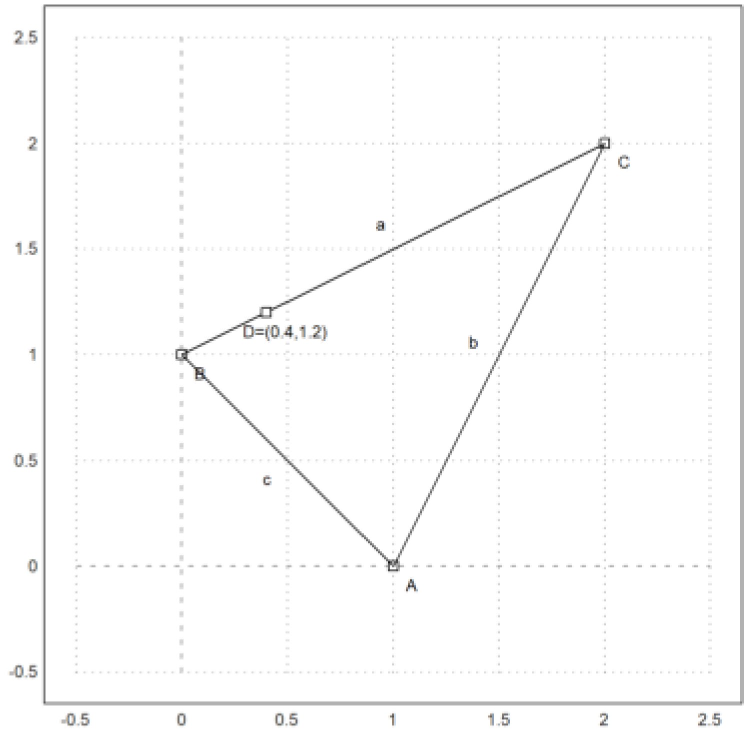
```
36°52'11.63''
```



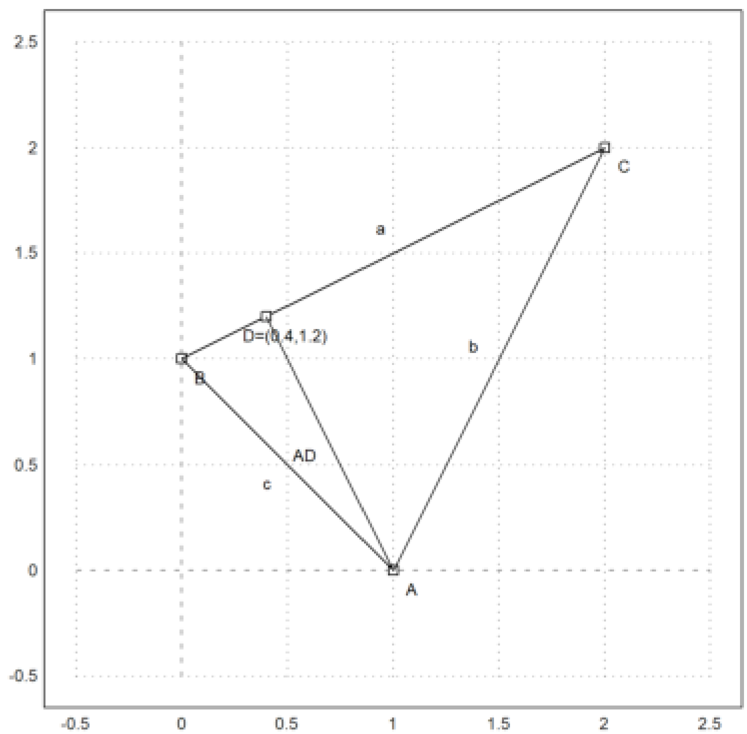
Gambar 408: images/GEOMETRI_Diva%20Nagita_(23030630024)-008.png



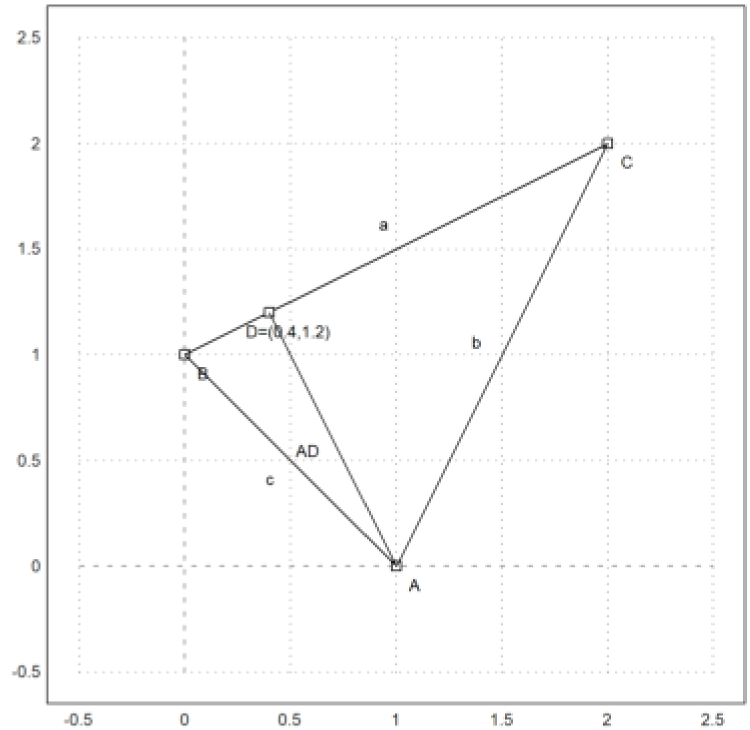
Gambar 409: images/GEOMETRI_Diva%20Nagita_(23030630024)-009.png



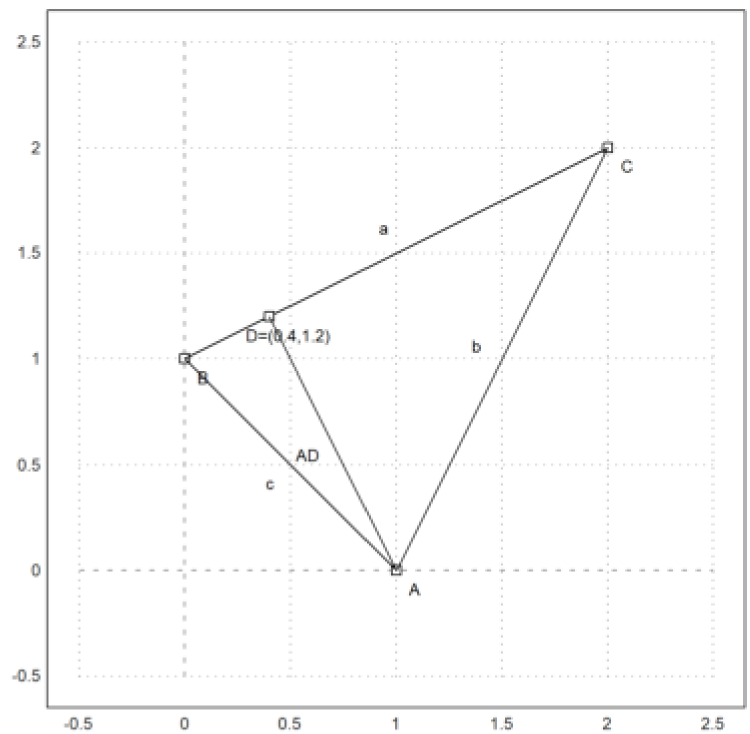
Gambar 410: images/GEOMETRI_Diva%20Nagita_(23030630024)-010.png



Gambar 411: images/GEOMETRI_Diva%20Nagita_(23030630024)-011.png

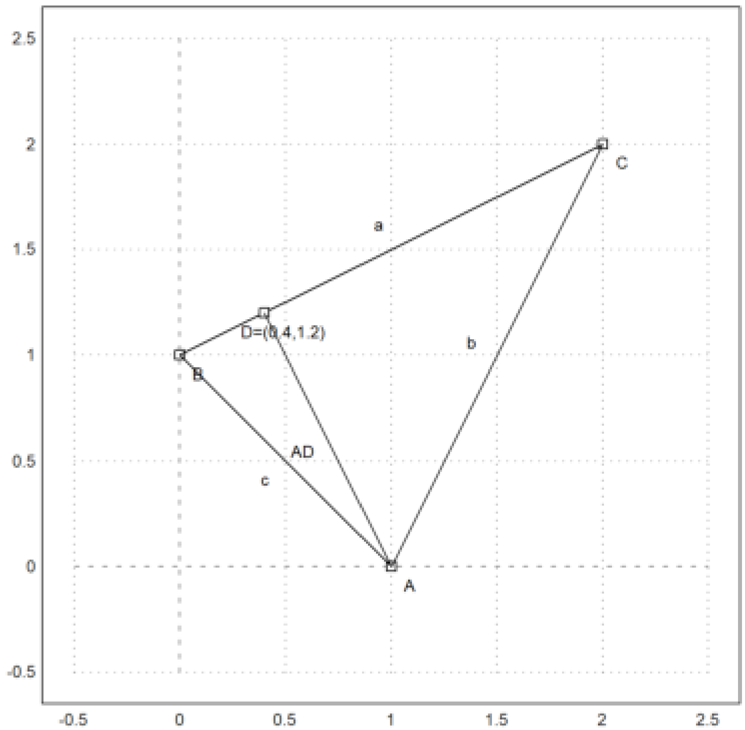


Gambar 412: images/GEOMETRI_Diva%20Nagita_(23030630024)-013.png



Gambar 413: images/GEOMETRI_Diva%20Nagita_(23030630024)-014.png

Sekarang, lingkarilah segitiga tersebut.
>c=circleThrough(A,B,C): // lingkaran luar segitiga ABC
>R=getCircleRadius(c): // jari2 lingkaran luar



Gambar 414: images/GEOMETRI_Diva%20Nagita_(23030630024)-015.png

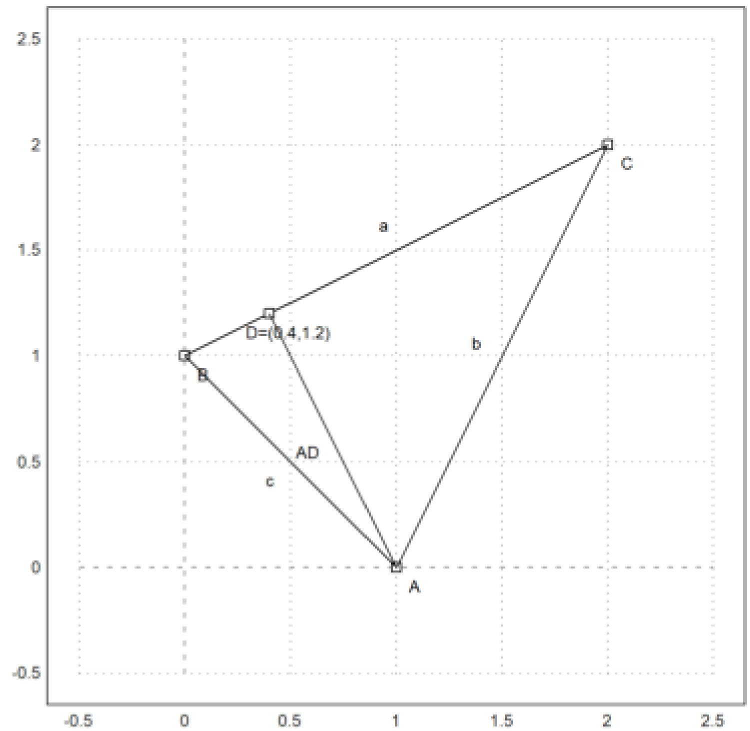
>O=getCircleCenter(c): // titik pusat lingkaran c
>plotPoint(O,"O"): // gambar titik "O"
>plotCircle(c,"Lingkaran luar segitiga ABC"):
Tampilkan koordinat titik pusat dan jari-jari lingkaran luar.
>O, R

```
[1.16667,  1.16667]
1.17851130198
```

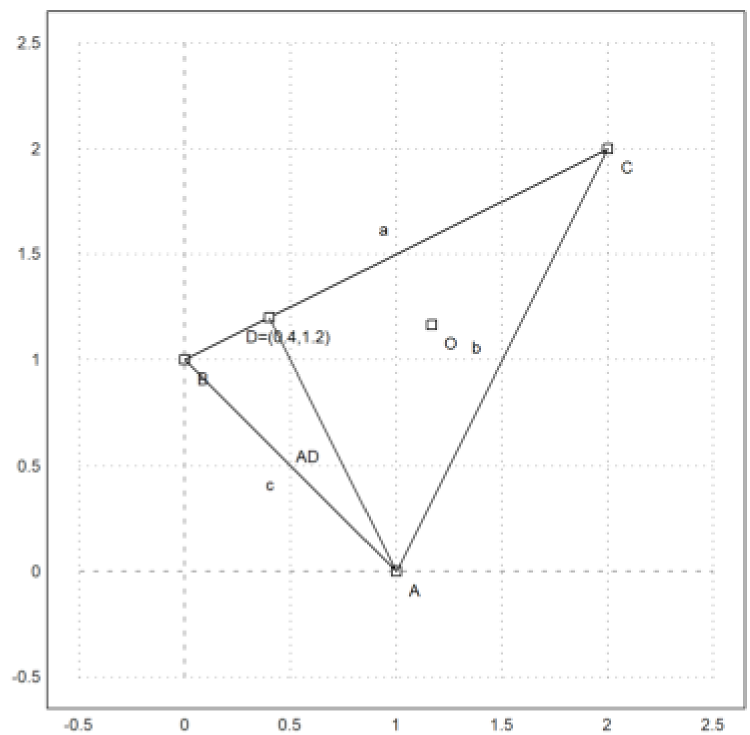
Sekarang akan digambar lingkaran dalam segitiga ABC. Titik pusat lingkaran dalam adalah titik potong garis-garis bagi sudut.
>l=angleBisector(A,C,B): // garis bagi <ACB
>g=angleBisector(C,A,B): // garis bagi <CAB
>P=lineIntersection(l,g): // titik potong kedua garis bagi sudut
Tambahkan semuanya ke plot.
>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
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"): // gambar lingkaran dalam

Latihan

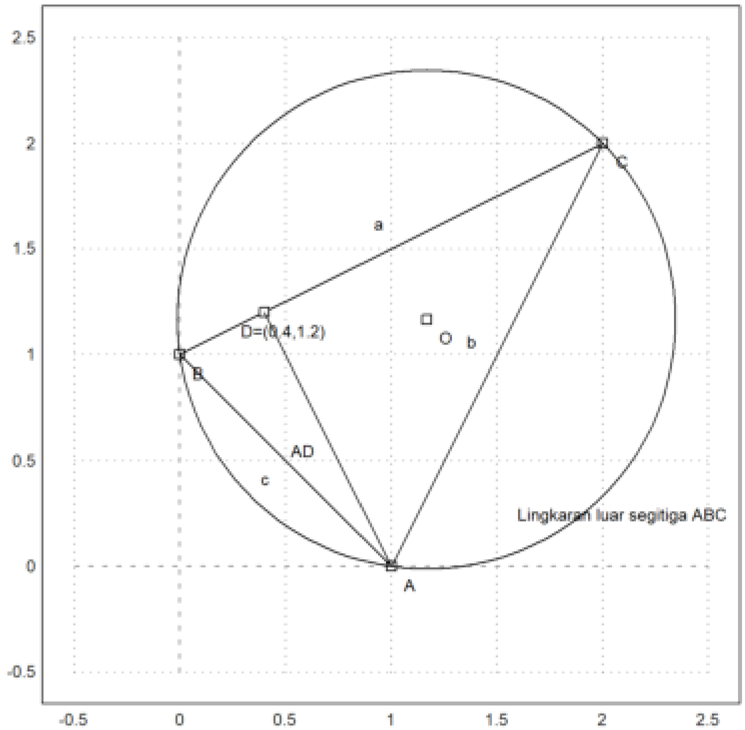
1. Tentukan ketiga titik singgung lingkaran dalam dengan sisi-sisi segitiga ABC.



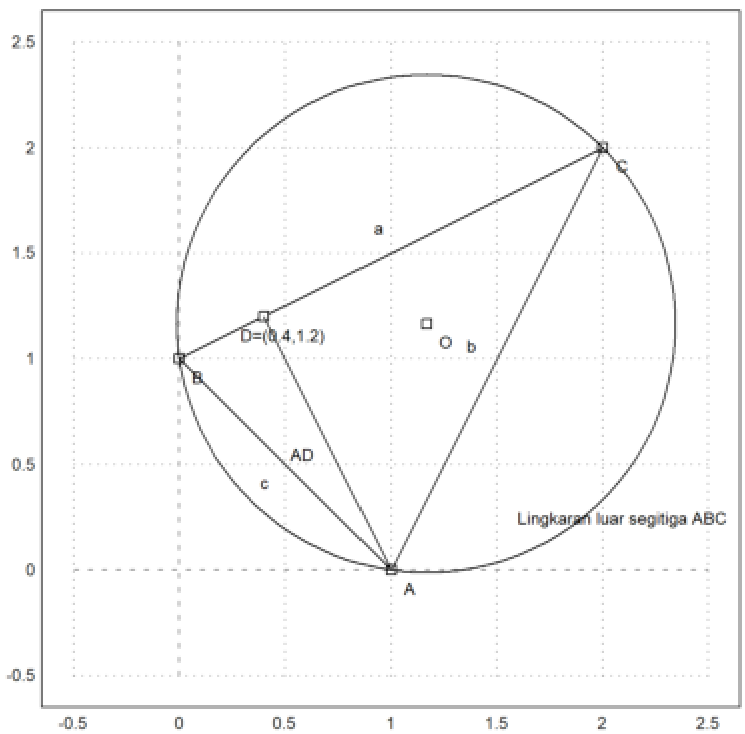
Gambar 415: images/GEOMETRI_Diva%20Nagita_(23030630024)-016.png



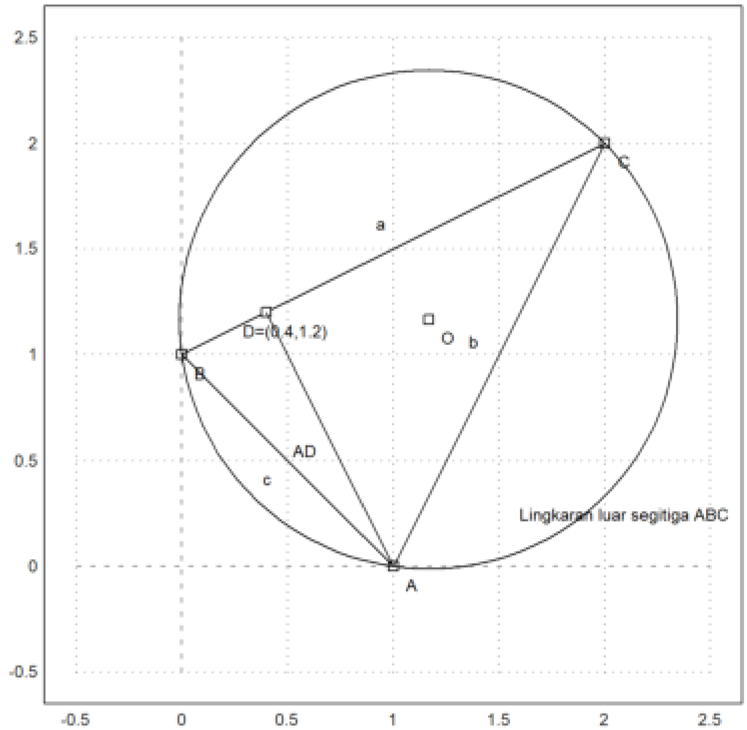
Gambar 416: images/GEOMETRI_Diva%20Nagita_(23030630024)-017.png



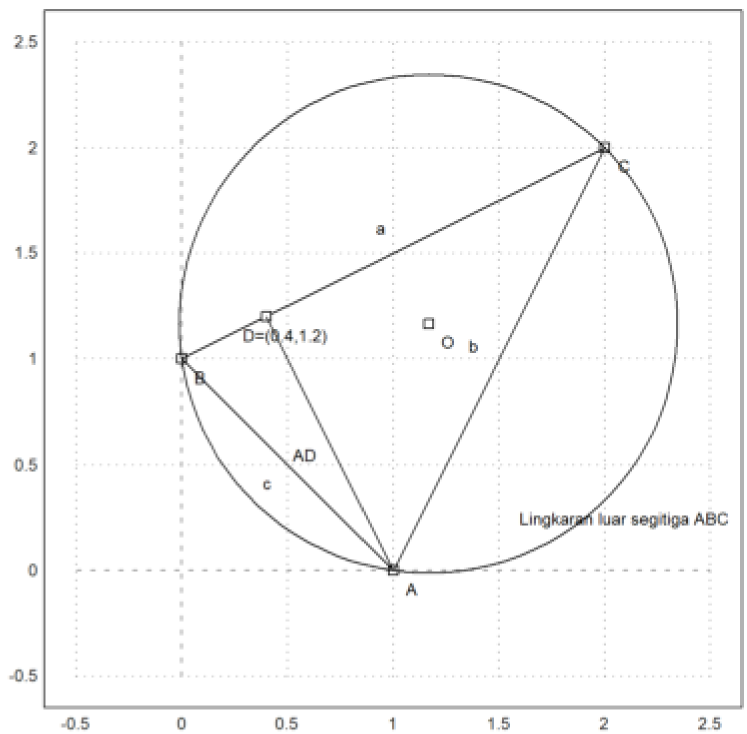
Gambar 417: images/GEOMETRI_Diva%20Nagita_(23030630024)-018.png



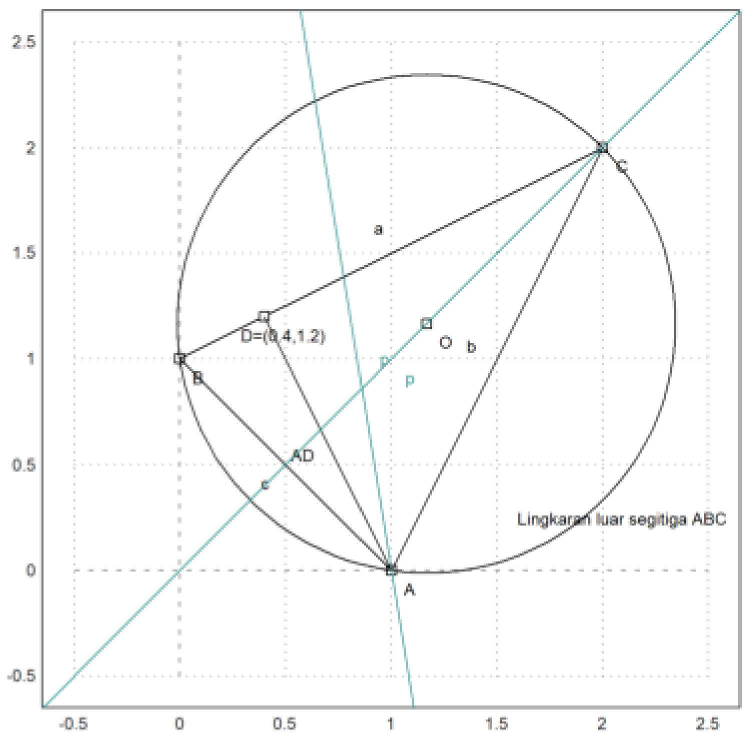
Gambar 418: images/GEOMETRI_Diva%20Nagita_(23030630024)-019.png



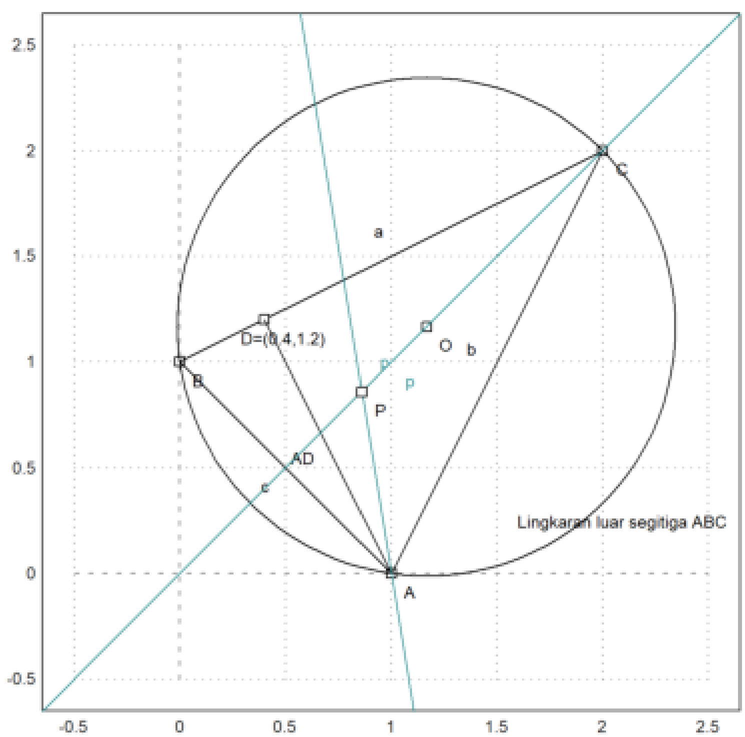
Gambar 419: images/GEOMETRI_Diva%20Nagita_(23030630024)-020.png



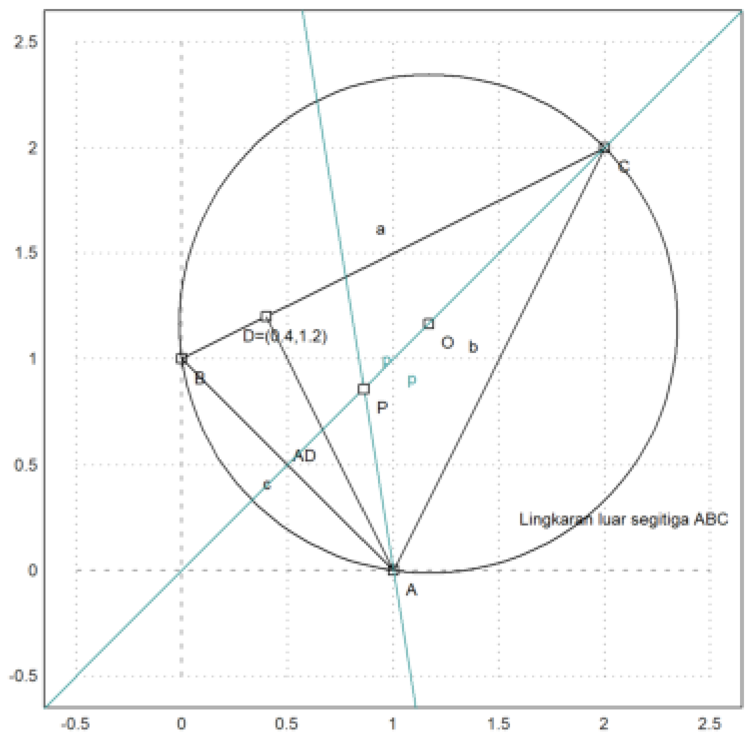
Gambar 420: images/GEOMETRI_Diva%20Nagita_(23030630024)-021.png



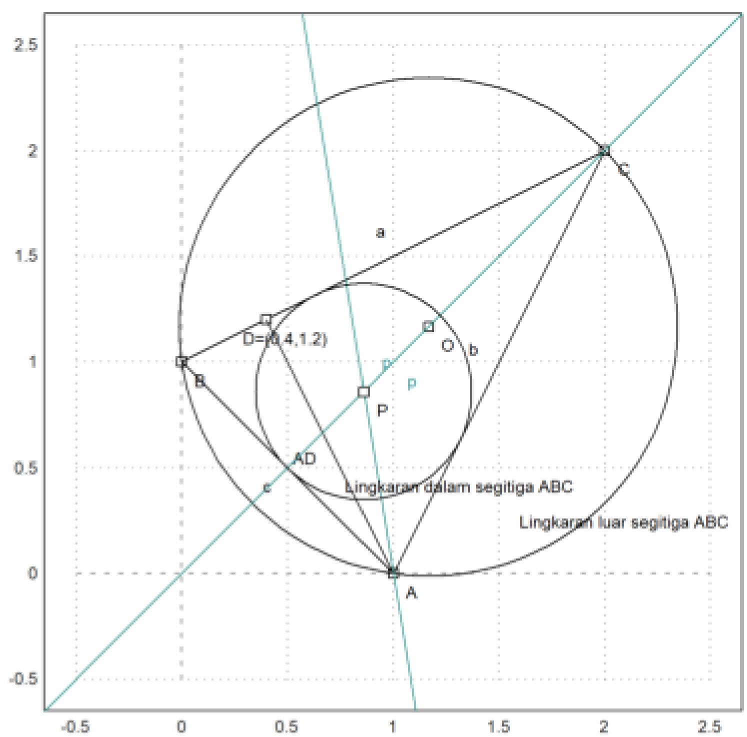
Gambar 421: images/GEOMETRI_Diva%20Nagita_(23030630024)-022.png



Gambar 422: images/GEOMETRI_Diva%20Nagita_(23030630024)-023.png

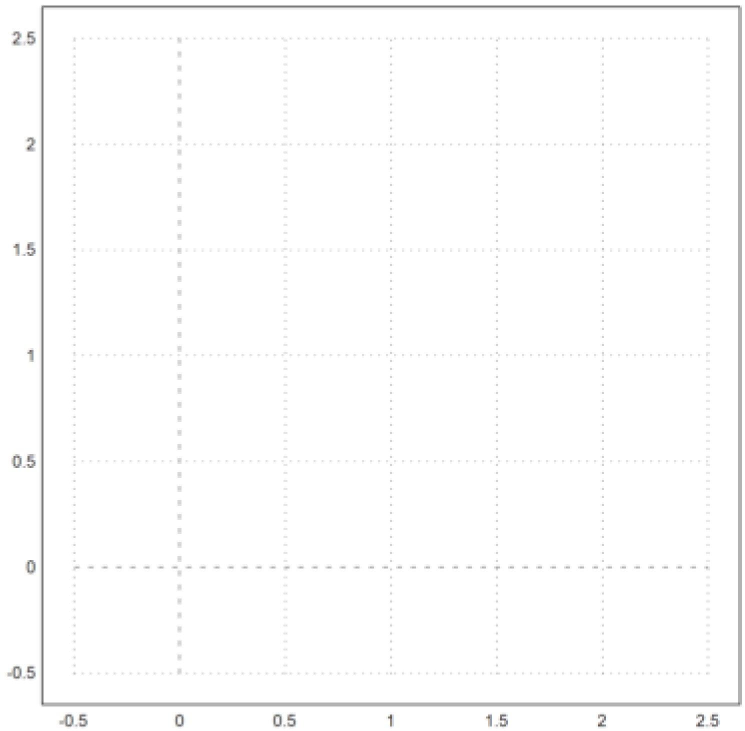


Gambar 423: images/GEOMETRI_Diva%20Nagita_(23030630024)-024.png



Gambar 424: images/GEOMETRI_Diva%20Nagita_(23030630024)-025.png

>setPlotRange(-0.5,2.5,-0.5,2.5):



Gambar 425: images/GEOMETRI_Diva%20Nagita_(23030630024)-026.png

>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"):
>K = lineCircleIntersections(lineThrough(A,B),circleWithCenter(P,r))

```
[0.5,  0.5]
```

>L = lineCircleIntersections(lineThrough(C,B),circleWithCenter(P,r))

```
[0.632456,  1.31623]
```

>M = lineCircleIntersections(lineThrough(C,A),circleWithCenter(P,r))

```
[1.31623,  0.632456]
```

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

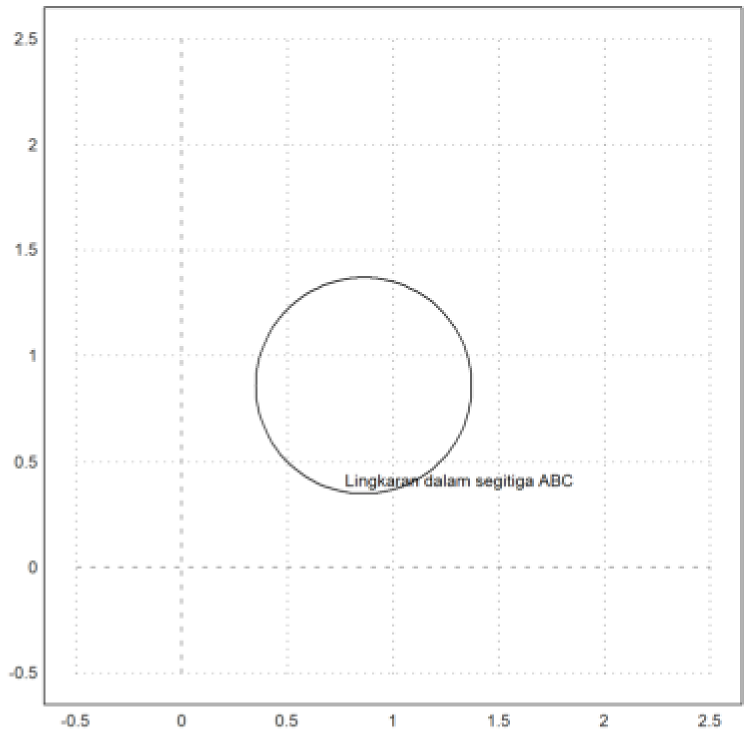
>plotPoint(K,"K"); plotPoint(M,"M"); plotPoint(L,"L");
>plotSegment(K,L,"m");
>plotSegment(L,M,"n");
>plotSegment(M,K,"l"):
>distance(P,lineIntersection(lineThrough(A,B),l))

```
0.509653732104
```

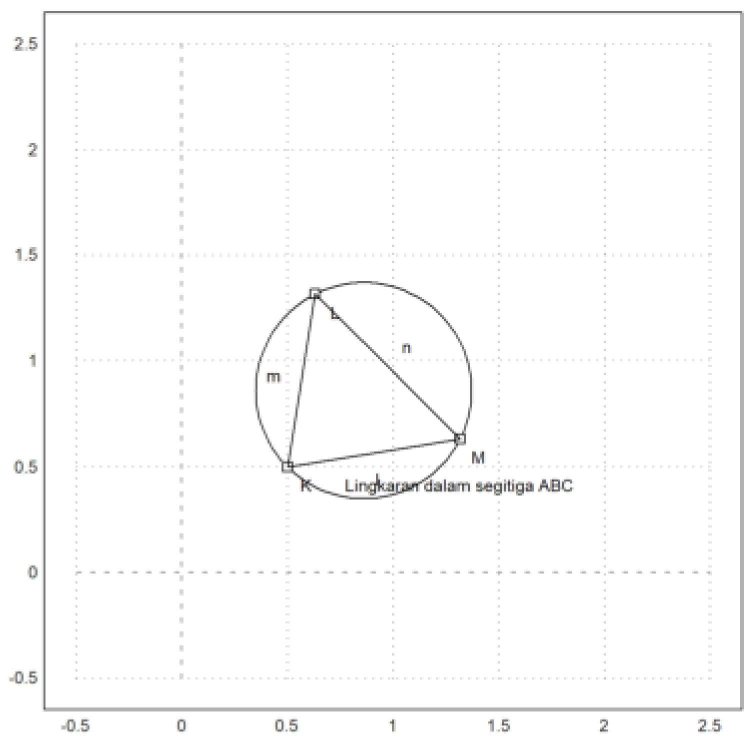
>getCircleRadius(circleWithCenter(P,r))

```
0.509653732104
```

>reset



Gambar 426: images/GEOMETRI_Diva%20Nagita_(23030630024)-027.png



Gambar 427: images/GEOMETRI_Diva%20Nagita_(23030630024)-028.png

```
0
```

>s=lineThrough(B,C)

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

>m=circleWithCenter(P,r)

```
[0.86038,  0.86038,  0.509654]
```

>S=lineCircleIntersections(s,m)

```
[0.632456,  1.31623]
```

Titik singgung garis AC dengan lingkaran dalam.
>p=lineThrough(A,C)

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

>Q=lineCircleIntersections(p,m)

```
[1.31623,  0.632456]
```

Titik singgung garis AB dengan lingkaran dalam.
>q=lineThrough(A,B)

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

>L=lineCircleIntersections(q,m)

```
[0.5,  0.5]
```

Jadi titik singgung lingkaran dalam dengan sisi-sisi segitiga adalah (0.632456, 1.31623), (1.31632, 0.632456), (0.5, 0.5).

2. Gambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut.

>setPlotRange(-0.5,2.5,-0.5,2.5);
>plotSegment(S,Q,"a"):
>plotSegment(S,L,"b"):
>plotSegment(L,Q,"c"):

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

>P, r

```
[0.86038,  0.86038]
0.509653732104
```

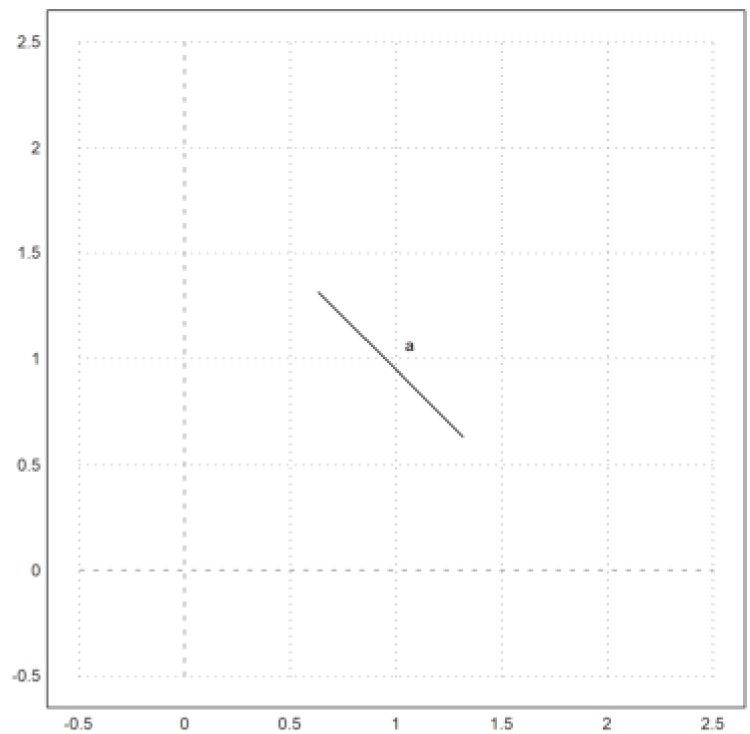
>k=angleBisector(A,B,C)

```
[-0.264911,  -1.63246,  -1.63246]
```

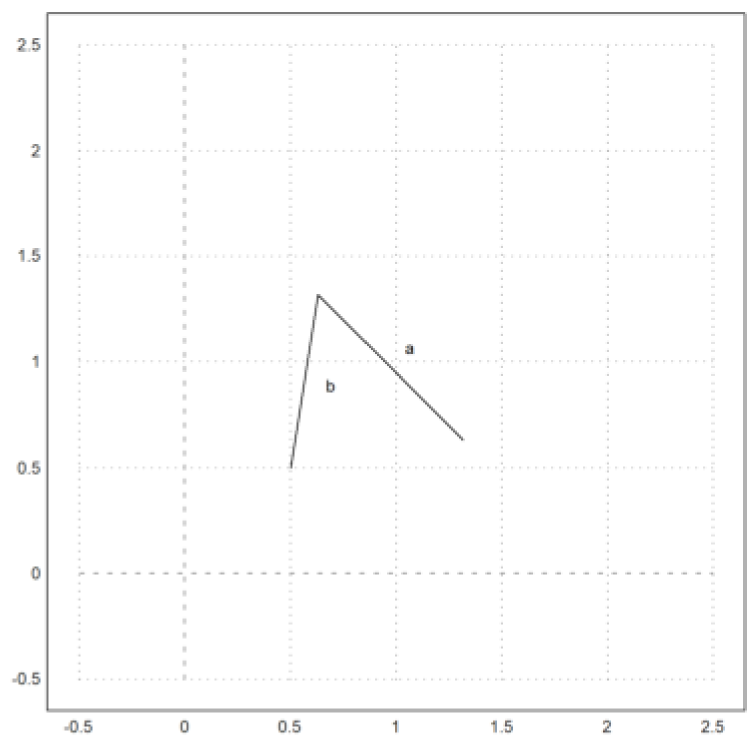
>color(2); plotLine(k):

4. Gambar jari-jari lingkaran dalam.

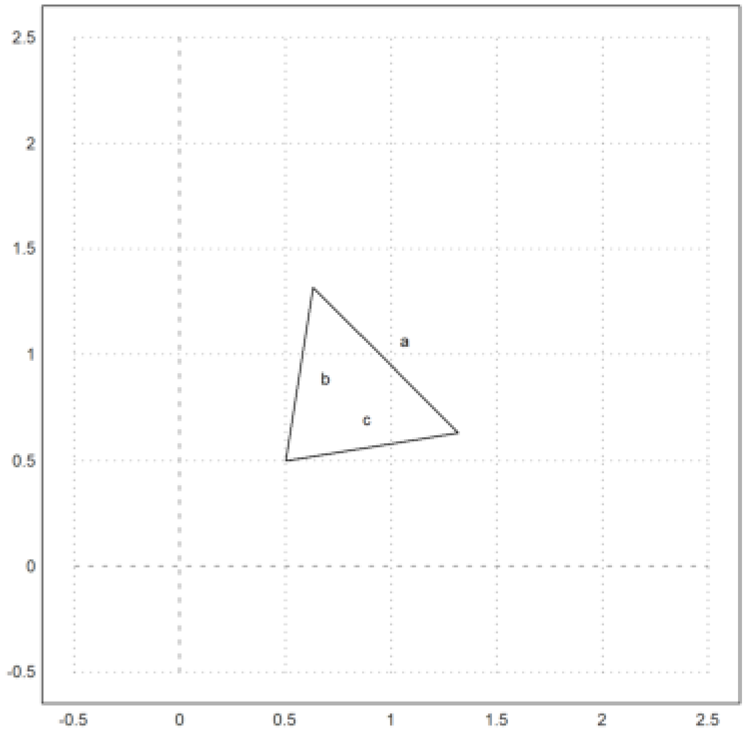
>plotSegment(P,L,"r"):



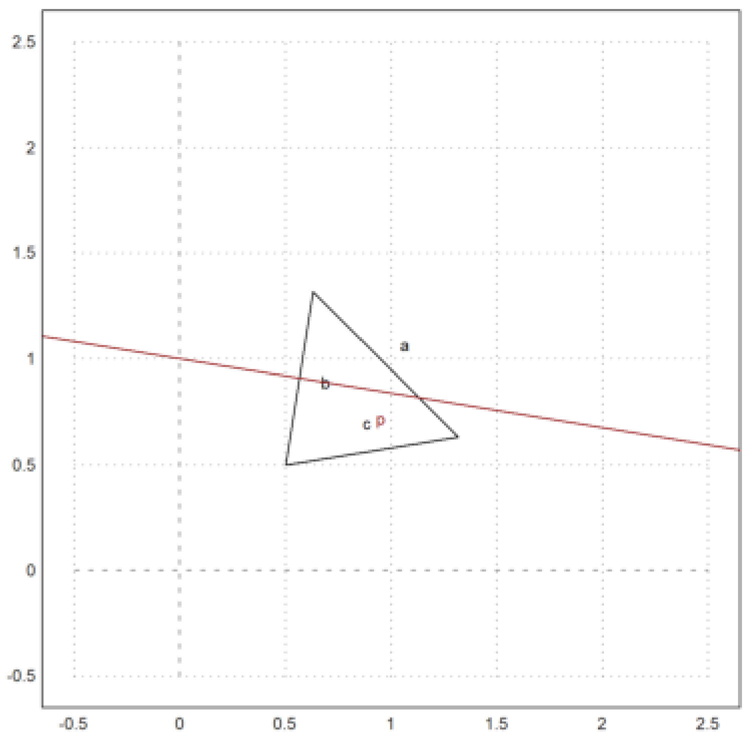
Gambar 428: images/GEOMETRI_Diva%20Nagita_(23030630024)-029.png



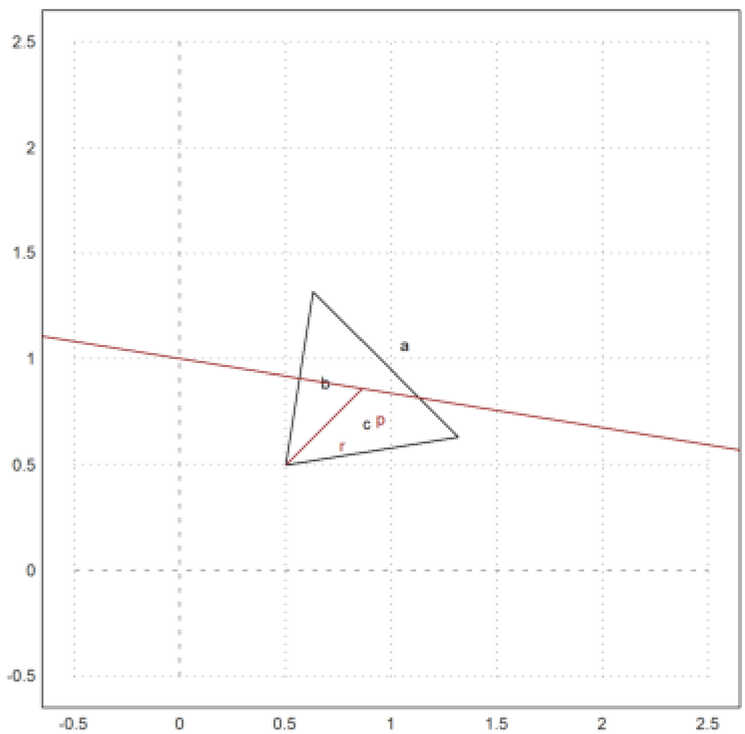
Gambar 429: images/GEOMETRI_Diva%20Nagita_(23030630024)-030.png



Gambar 430: images/GEOMETRI_Diva%20Nagita_(23030630024)-031.png



Gambar 431: images/GEOMETRI_Diva%20Nagita_(23030630024)-032.png



Gambar 432: images/GEOMETRI_Diva%20Nagita_(23030630024)-033.png

Geometri Smbolik

Kita dapat menghitung geometri eksak dan simbolik dengan menggunakan Maxima.

File geometri.e menyediakan fungsi-fungsi yang sama (dan lebih banyak lagi) di Maxima. Namun, kita dapat menggunakan komputasi simbolik 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 menyediakan komputasi simbolis.

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

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

Kita bisa mendapatkan persamaan untuk sebuah garis dengan mudah.

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

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

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

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

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

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

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

$$(x_1 - 1)\ y - x\ y_1 = -y_1$$

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

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

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

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

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

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

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

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

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

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

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

$$1.178511301977579$$

1.178511301977579

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

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

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

$$y = x$$

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

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

>P() // hasilnya sama dengan perhitungan sebelumnya

```
[0.86038,  0.86038]
```

Garis dan Lingkaran yang Berpotongan

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

>A &:= [1,0]; c=circleWithCenter(A,4);

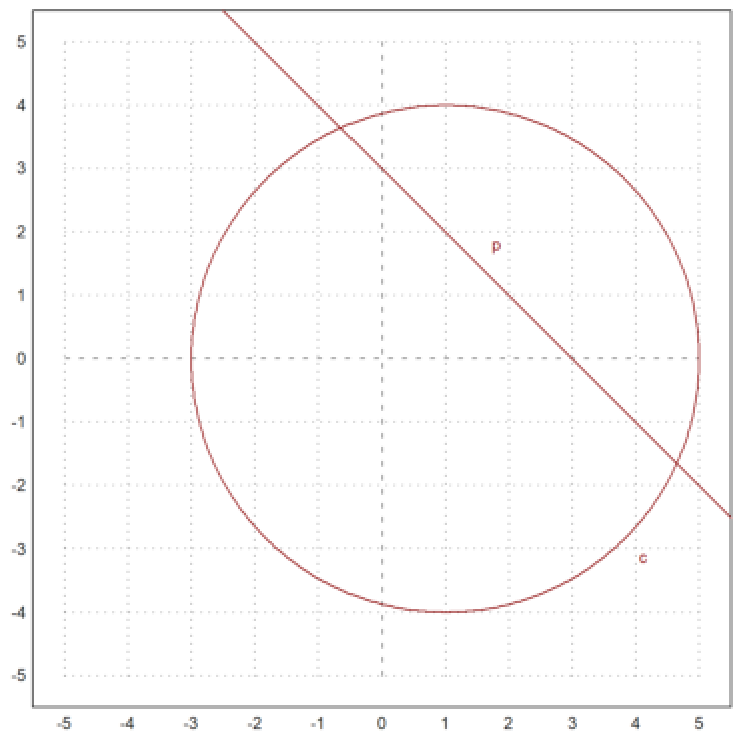
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);

>setPlotRange(5); plotCircle(c); plotLine(l):

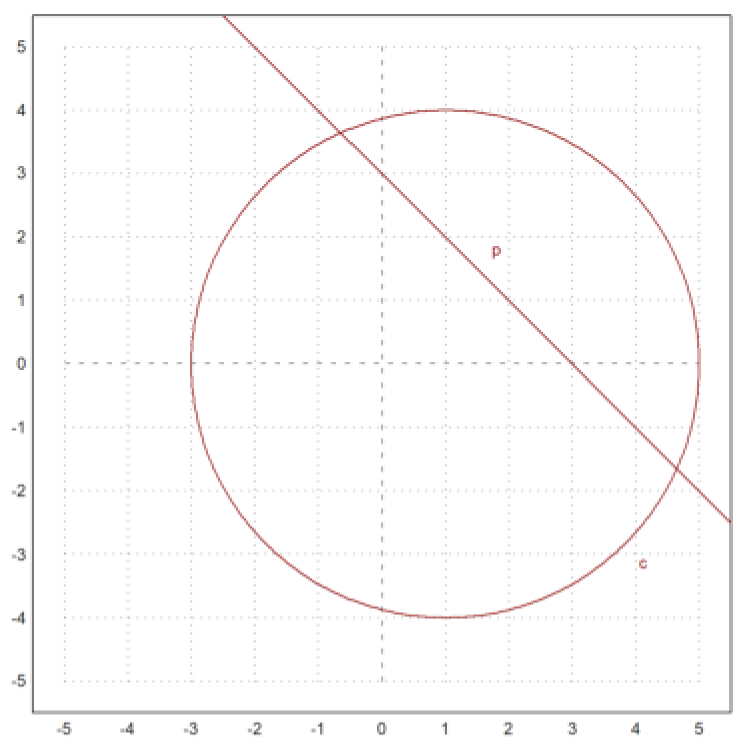
The intersection of a line with a circle returns two points and the number of intersection points.

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

>P1, P2, f



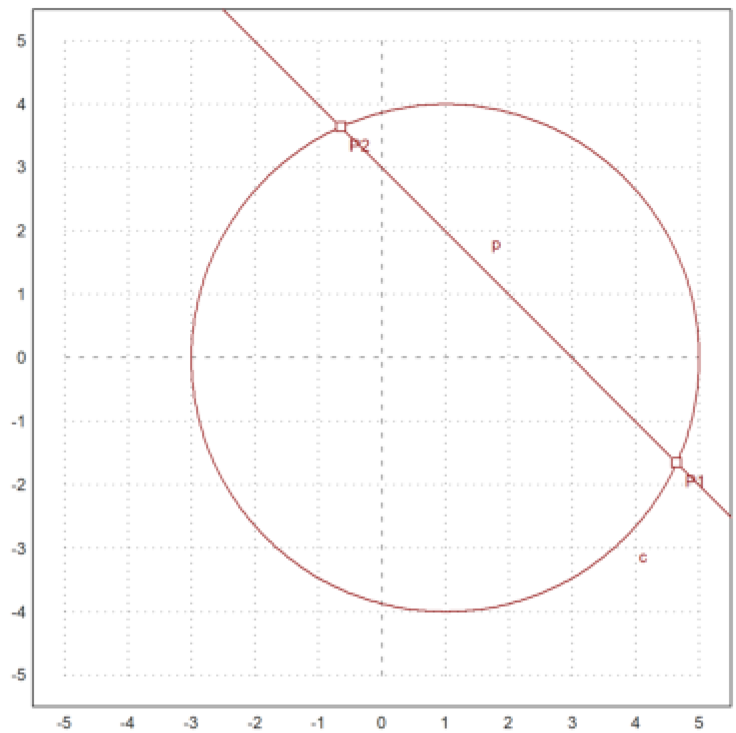
Gambar 433: images/GEOMETRI_Diva%20Nagita_(23030630024)-047.png



Gambar 434: images/GEOMETRI_Diva%20Nagita_(23030630024)-048.png

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

>plotPoint(P1); plotPoint(P2):



Gambar 435: images/GEOMETRI_Diva%20Nagita_(23030630024)-049.png

Hal yang sama pada Maxima.

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

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

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

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

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

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

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

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

>degprint(computeAngle(P1,C,P2))

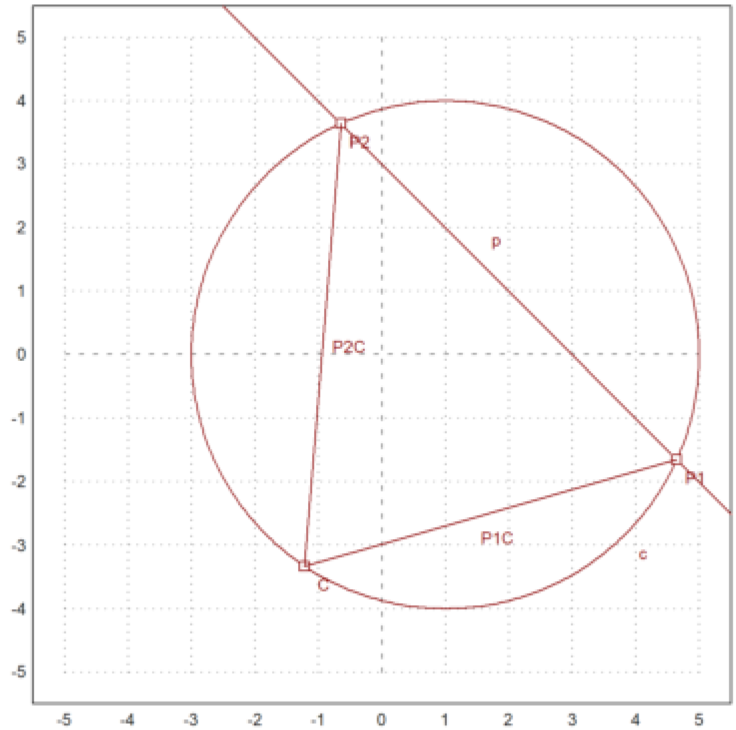
```
69°17'42.68''
```

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

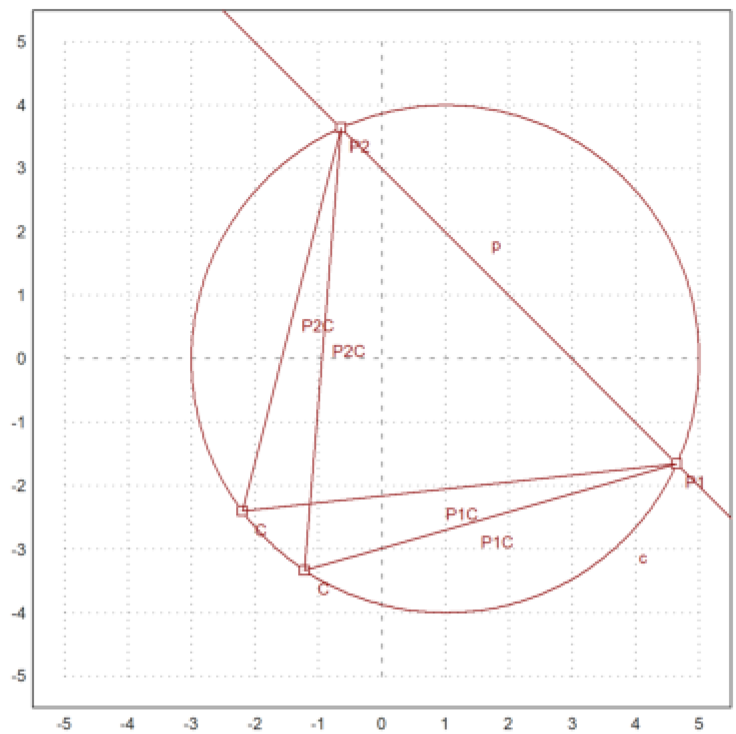
>degprint(computeAngle(P1,C,P2))

```
69°17'42.68''
```

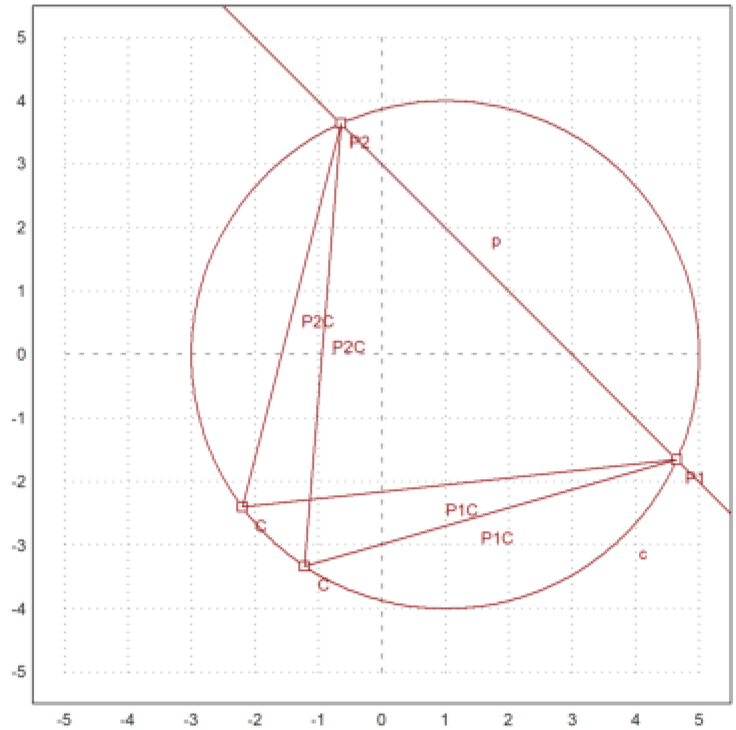
>insimg;



Gambar 436: images/GEOMETRI_Diva%20Nagita_(23030630024)-051.png



Gambar 437: images/GEOMETRI_Diva%20Nagita_(23030630024)-052.png



Gambar 438: images/GEOMETRI_Diva%20Nagita_(23030630024)-053.png

Garis Sumbu

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

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

>A=[2,2]; B=[-1,-2];

>c1=circleWithCenter(A,distance(A,B));

>c2=circleWithCenter(B,distance(A,B));

>{P1,P2,f}=circleCircleIntersections(c1,c2);

>l=lineThrough(P1,P2);

>setPlotRange(5); plotCircle(c1); plotCircle(c2):

>plotPoint(A); plotPoint(B); plotSegment(A,B); plotLine(l):

Selanjutnya, kami melakukan hal yang sama di Maxima dengan koordinat umum.

>A &= [a1,a2]; B &= [b1,b2];

>c1 &= circleWithCenter(A,distance(A,B));

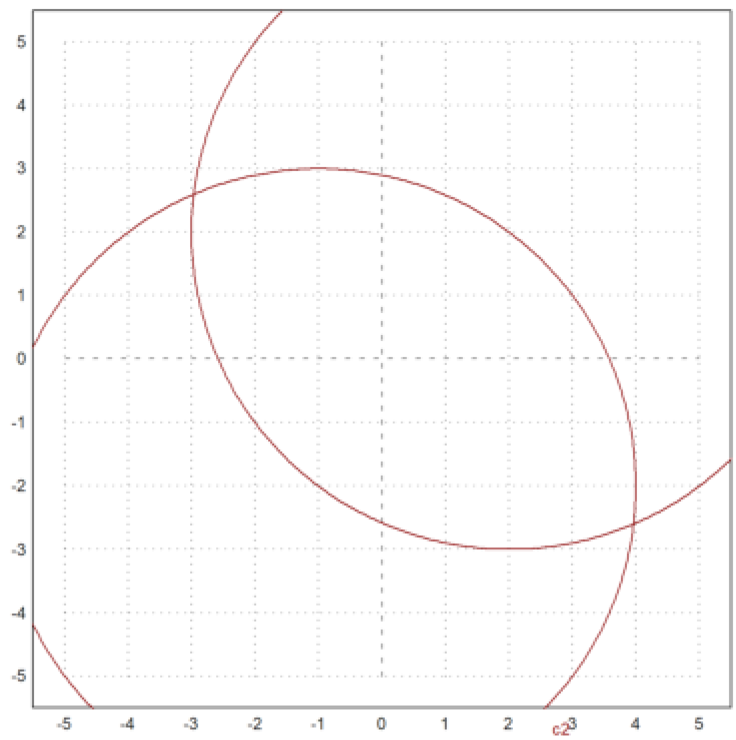
>c2 &= circleWithCenter(B,distance(A,B));

>P &= circleCircleIntersections(c1,c2); P1 &= P[1]; P2 &= P[2];

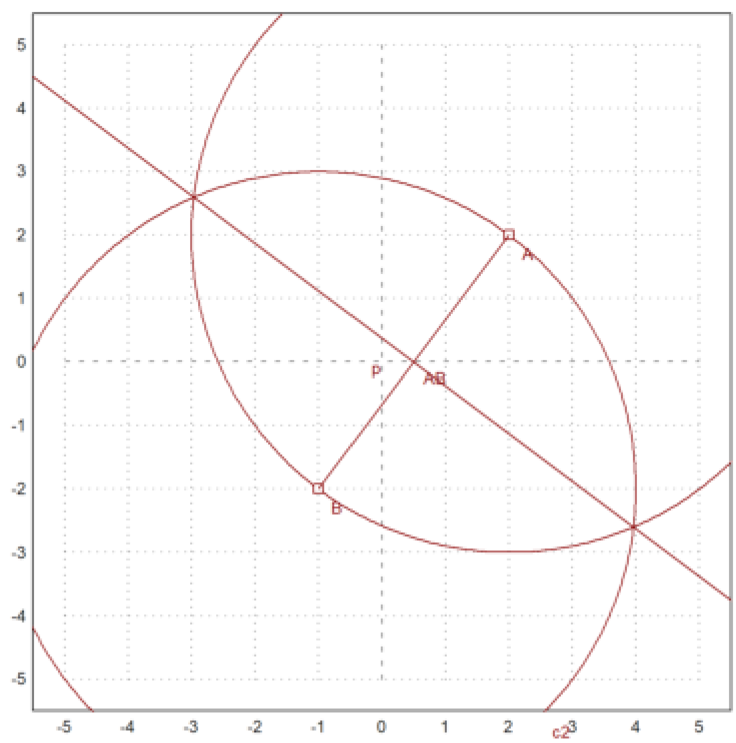
Persamaan untuk persimpangan cukup rumit. Tetapi kita dapat menyederhanakannya, jika kita menyelesaikan untuk y.

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

>$solve(g,y)



Gambar 439: images/GEOMETRI_Diva%20Nagita_(23030630024)-054.png



Gambar 440: images/GEOMETRI_Diva%20Nagita_(23030630024)-055.png

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

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.

Rumus Heron

Rumus Heron menyatakan bahwa luas segitiga dengan panjang sisi-sisi a, b dan c adalah:

$$L=\sqrt{s(s-a)(s-b)(s-c)}\quad \text{dengan } s=(a+b+c)/2,$$

atau bisa ditulis dalam bentuk lain:

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

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

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

Nilai y didapat dengan menyelesaikan sistem persamaan:

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

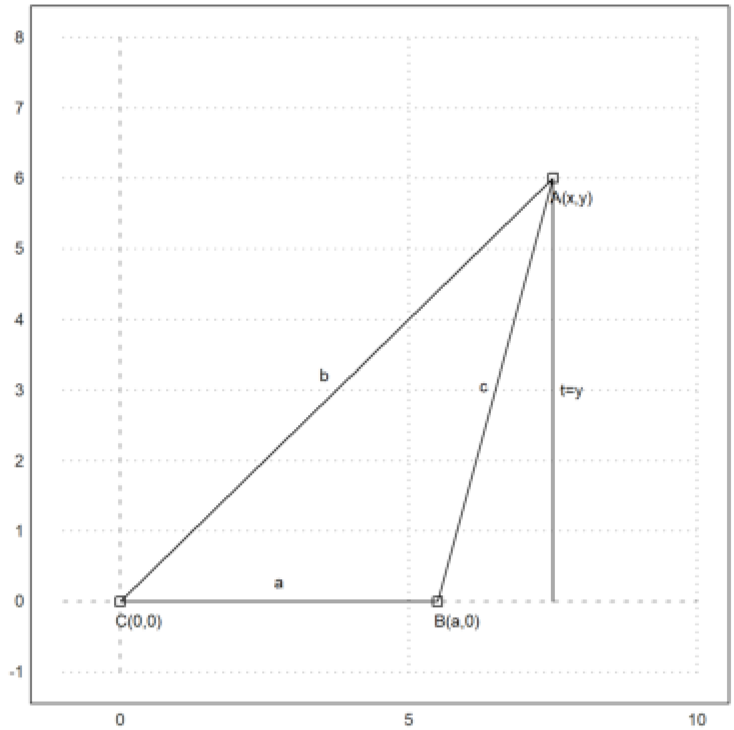
>setPlotRange(-1,10,-1,8); plotPoint([0,0], “C(0,0)”); plotPoint([5.5,0], “B(a,0)”); . . .
> plotPoint([7.5,6], “A(x,y)”);

>plotSegment([0,0],[5.5,0], “a”,25); plotSegment([5.5,0],[7.5,6],“c”,15); . . .
> plotSegment([0,0],[7.5,6],“b”,25);

>plotSegment([7.5,6],[7.5,0],“t=y”,25):

>&assume(a>0); sol &= solve([x^{2+y}2=b$^{2,(x-a)}$2+y$^{2=c}$2],[x,y])

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



Gambar 441: images/GEOMETRI_Diva%20Nagita_(23030630024)-064.png

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

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

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

Kami mendapatkan rumus Heron.
>function H(a,b,c) &= sqrt(factor((ysol*a/2)^2)); $'H(a,b,c)=H(a,b,c)

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

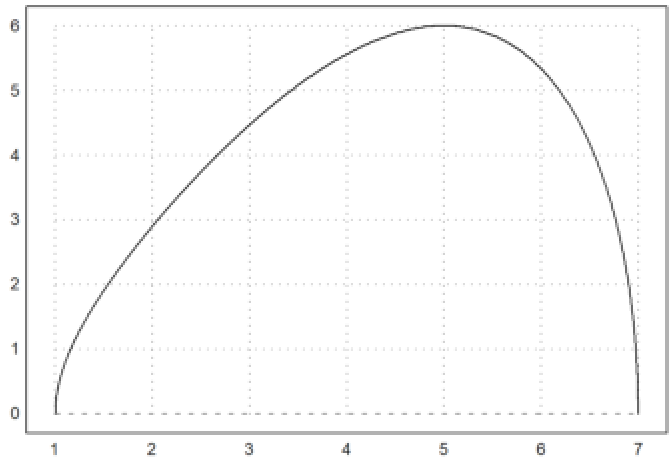
>$'Luas=H(2,5,6) // luas segitiga dengan panjang sisi-sisi 2, 5, 6

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

Tentu saja, setiap segitiga persegi panjang adalah kasus yang terkenal.
>H(3,4,5) //luas segitiga siku-siku dengan panjang sisi 3, 4, 5

```
6
```

Dan juga jelas, bahwa ini adalah segitiga dengan luas maksimal dan kedua sisi 3 dan 4.



Gambar 442: images/GEOMETRI_Diva%20Nagita_(23030630024)-068.png

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

Kasus umum juga bisa digunakan.

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

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

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

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

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

Dan membuat fungsi-fungsi ini.

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

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

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

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

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

Kita dapat memeriksa persamaan umum untuk elips ini, yaitu

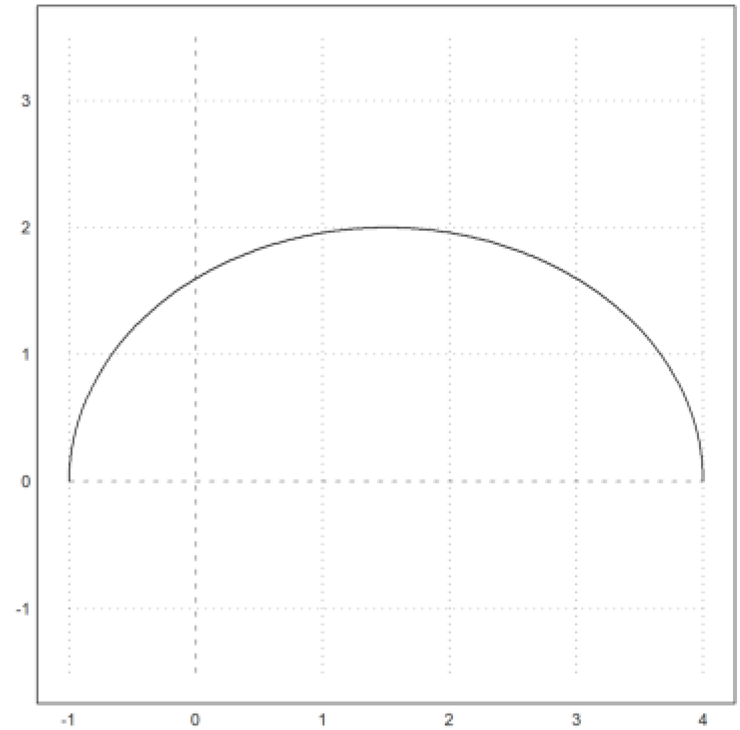
$$\frac{(x-x_m)^2}{u^2}+\frac{(y-y_m)}{v^2}=1,$$

di mana (xm, ym) adalah pusat, dan u dan v adalah setengah sumbu.

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

$$1$$

Kita melihat bahwa tinggi dan luas segitiga adalah maksimal untuk x=0. Dengan demikian, luas segitiga dengan a+b+c=d adalah maksimal, jika segitiga tersebut sama sisi. Kita ingin membuktikannya secara analitis.



Gambar 443: images/GEOMETRI_Diva%20Nagita_(23030630024)-073.png

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

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

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

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

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

Ada juga metode Lagrange, yang 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])

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

Kita bisa membuat plot situasi

Pertama-tama, tetapkan titik-titik di Maxima.

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

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

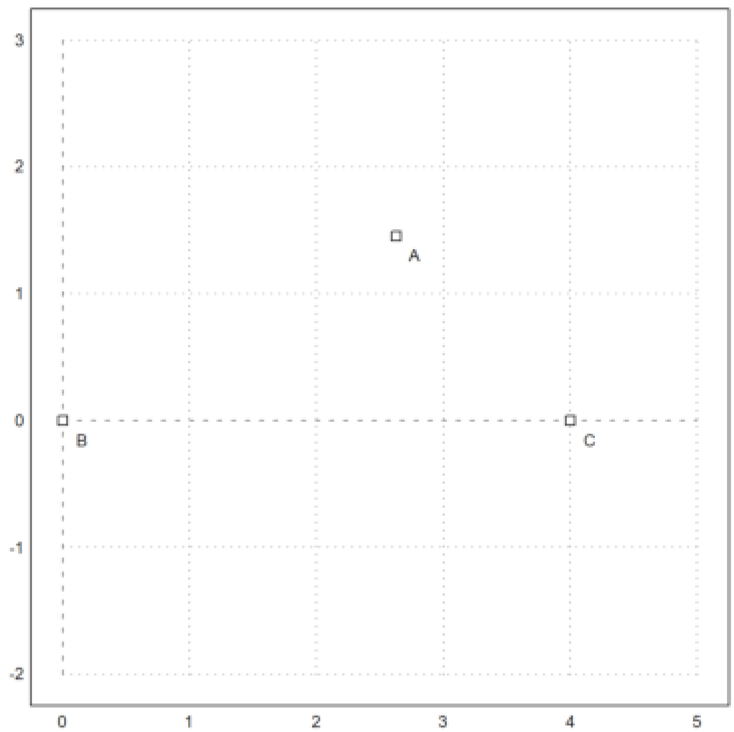
>B &= [0,0]; $B, C &= [a,0]; $C

$$[0,0]$$

$$[a,0]$$

Kemudian, tetapkan kisaran plot, dan plot titik-titiknya.

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



Gambar 444: images/GEOMETRI_Diva%20Nagita_(23030630024)-081.png

Plot segmen-segmen tersebut.

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

Hitung garis tegak lurus tengah dalam Maxima.

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

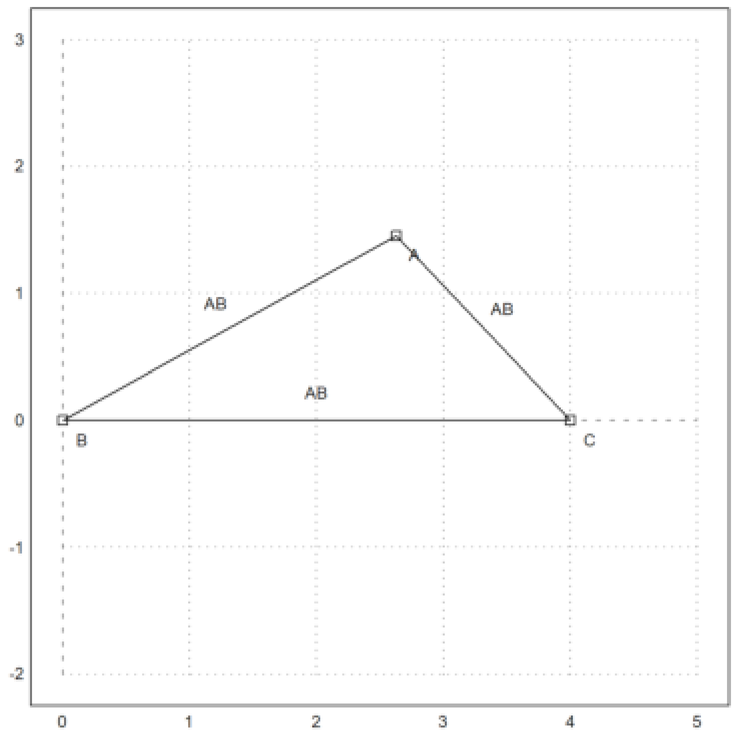
dan bagian tengah lingkar.

>U &= lineIntersection(h,g);

Kita mendapatkan rumus untuk jari-jari lingkaran.

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

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



Gambar 445: images/GEOMETRI_Diva%20Nagita_(23030630024)-082.png

Mari kita tambahkan ini ke dalam plot.

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

Dengan menggunakan geometri, kami memperoleh rumus sederhana

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

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

>$c^{2/\sin(\text{computeAngle(A,B,C)})}2 | factor

$$\frac{50\,c^2}{49}$$

Contoh 4: Garis Euler dan Parabola

Garis Euler adalah garis yang ditentukan dari segitiga apa pun yang tidak sama sisi. Garis ini merupakan garis tengah segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga, termasuk ortosentrum, circumcentrum, centroid, titik Exeter, dan pusat lingkaran sembilan titik segitiga.

Sebagai demonstrasi, kami menghitung dan memplot garis Euler dalam sebuah segitiga.

Pertama, kita mendefinisikan sudut-sudut segitiga dalam Euler. Kami menggunakan definisi, yang terlihat dalam ekspresi simbolis.

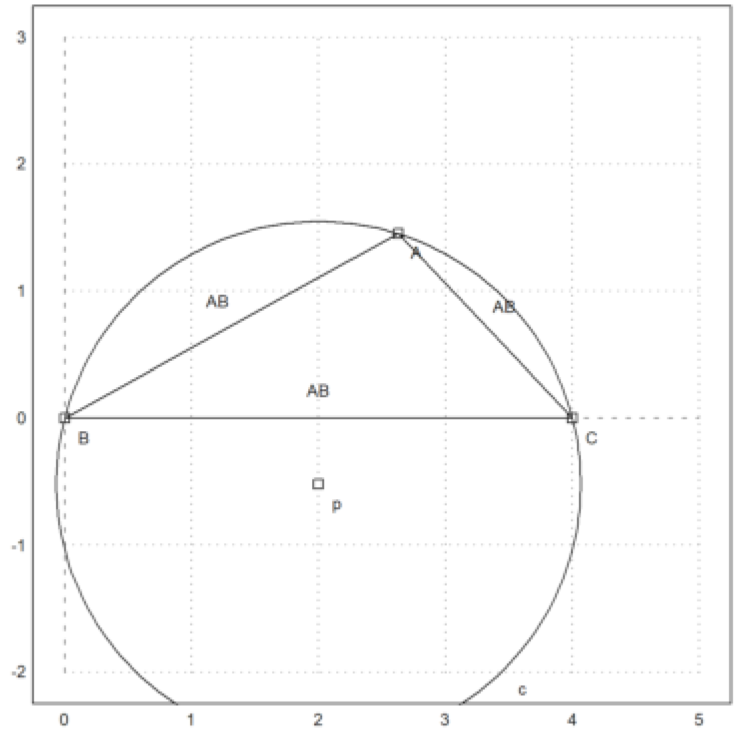
>A::=[-1,-1]; B::=[2,0]; C::=[1,2];

Untuk memplot objek geometris, kita menyiapkan area plot, dan menambahkan titik-titiknya. Semua plot objek geometris ditambahkan ke plot saat ini.

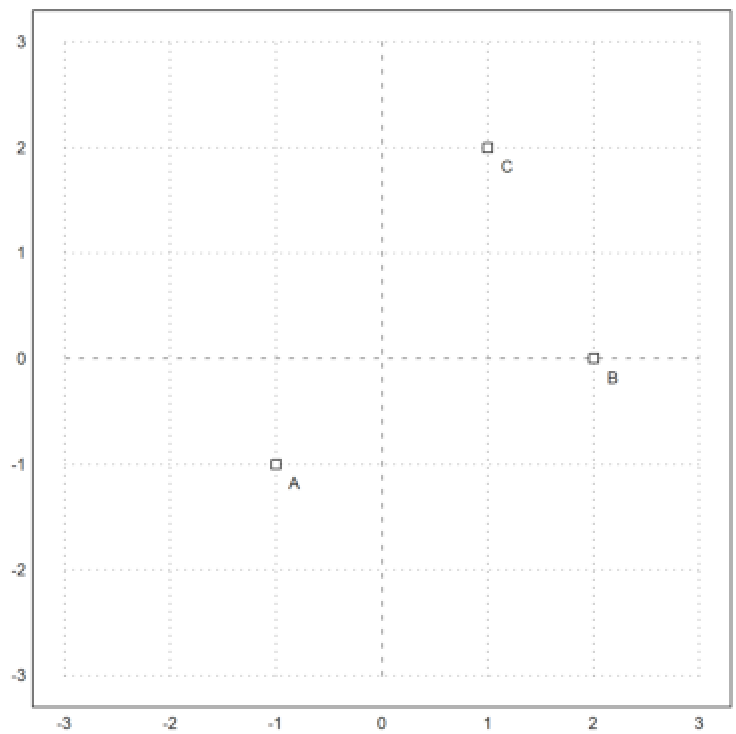
>setPlotRange(3); plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"):

Kita juga bisa menambahkan sisi-sisi segitiga.

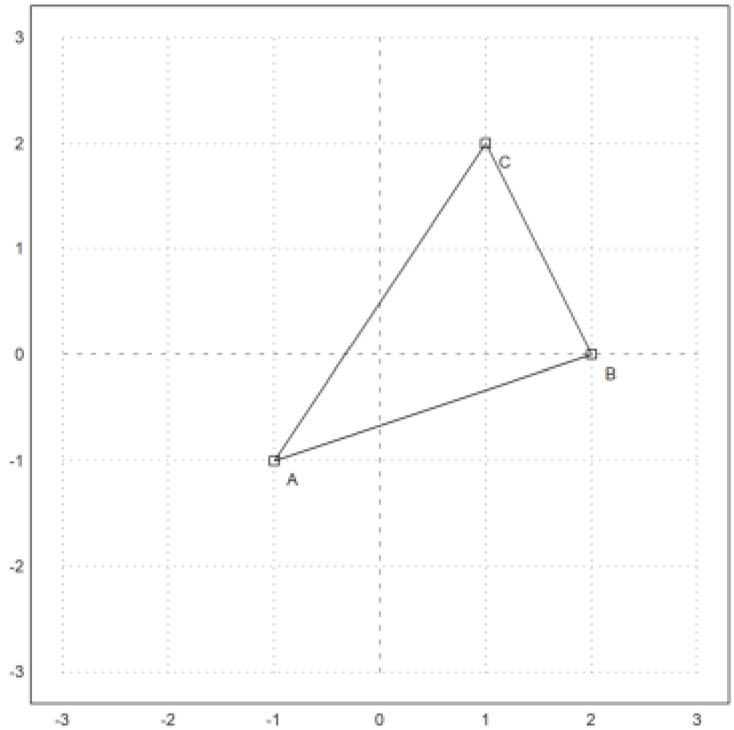
>plotSegment(A,B," "); plotSegment(B,C,""); plotSegment(C,A," "):



Gambar 446: images/GEOMETRI_Diva%20Nagita_(23030630024)-084.png



Gambar 447: images/GEOMETRI_Diva%20Nagita_(23030630024)-087.png



Gambar 448: images/GEOMETRI_Diva%20Nagita_(23030630024)-088.png

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

>$areaTriangle(A,B,C)

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

Kita dapat menghitung koefisien dari sisi c.

>c &= lineThrough(A,B)

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

dan juga mendapatkan formula untuk baris ini.

>$getLineEquation(c,x,y)

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

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

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

$$\frac{3\,y-x+2}{\sqrt{10}}$$
$$\frac{7}{\sqrt{10}}$$

Sekarang kita menghitung keliling ABC.

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

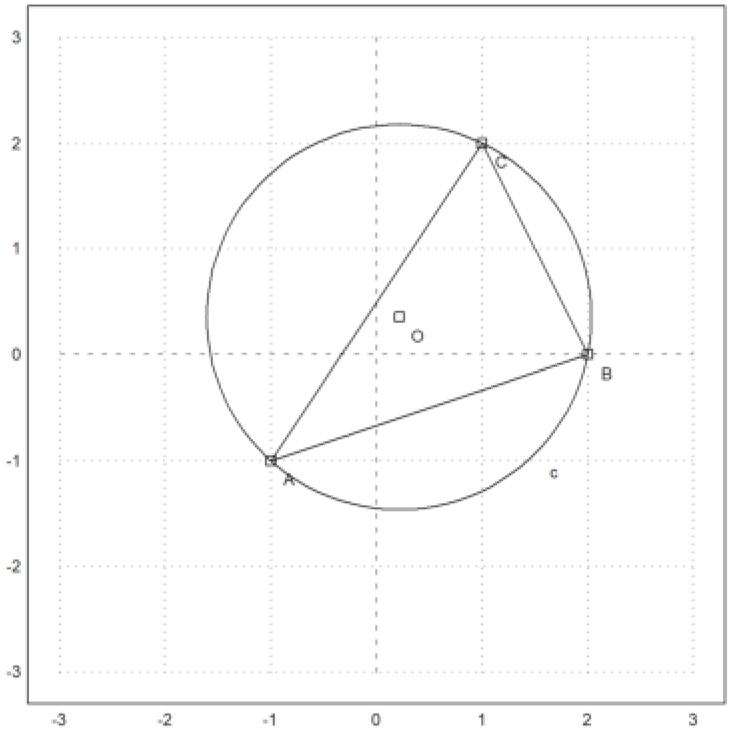
$$\left(y-\frac{5}{14}\right)^2+\left(x-\frac{3}{14}\right)^2=\frac{325}{98}$$

>O &= getCircleCenter(LL); $O

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

Plot lingkaran dan pusatnya. Cu dan U adalah simbolik. Kami mengevaluasi ekspresi ini untuk Euler.

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



Gambar 449: images/GEOMETRI_Diva%20Nagita_(23030630024)-095.png

Kita dapat menghitung perpotongan ketinggian di ABC (pusat ortosentrum) secara numerik dengan perintah berikut ini.

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

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

Sekarang kita dapat menghitung garis Euler dari segitiga tersebut.

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

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

Tambahkan ke plot kami.

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

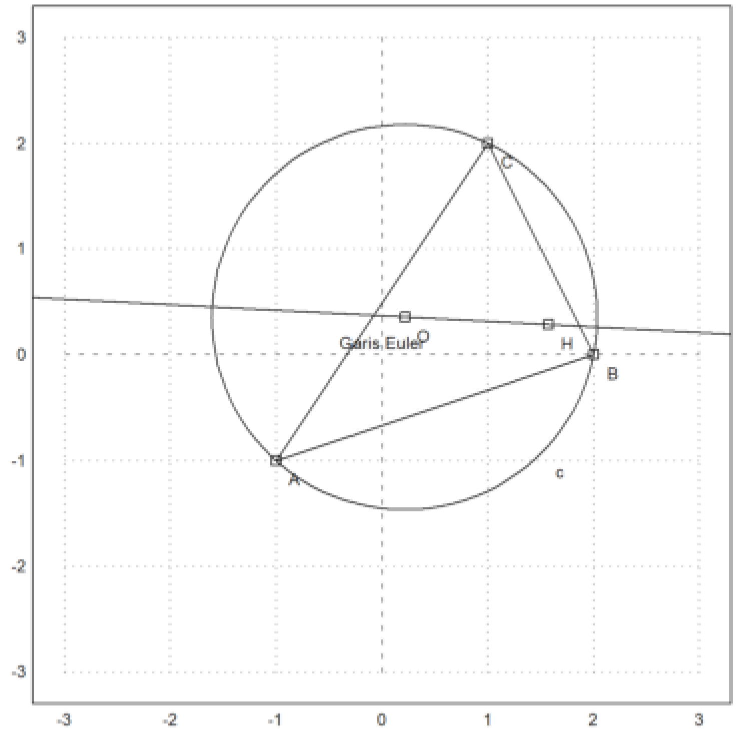
Pusat gravitasi harus berada pada garis ini.

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

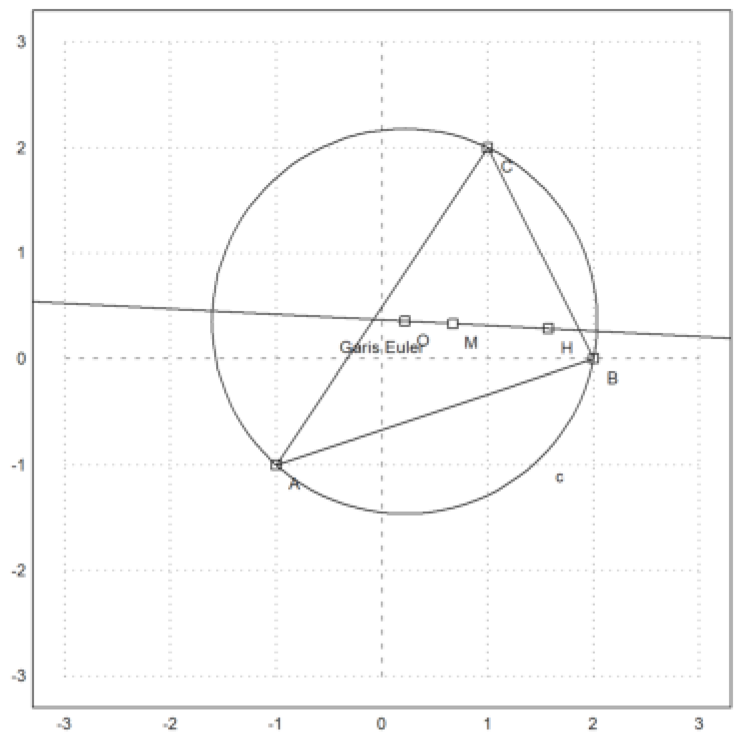
$$-\frac{1}{2} = -\frac{1}{2}$$

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

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



Gambar 450: images/GEOMETRI_Diva%20Nagita_(23030630024)-098.png



Gambar 451: images/GEOMETRI_Diva%20Nagita_(23030630024)-100.png

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

$$2$$

Fungsi-fungsi ini juga mencakup fungsi untuk sudut.

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

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

60°15’18.43’ ’

Persamaan untuk bagian tengah lingkaran tidak terlalu bagus.

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

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

Mari kita hitung juga ekspresi untuk jari-jari lingkaran yang tertulis.

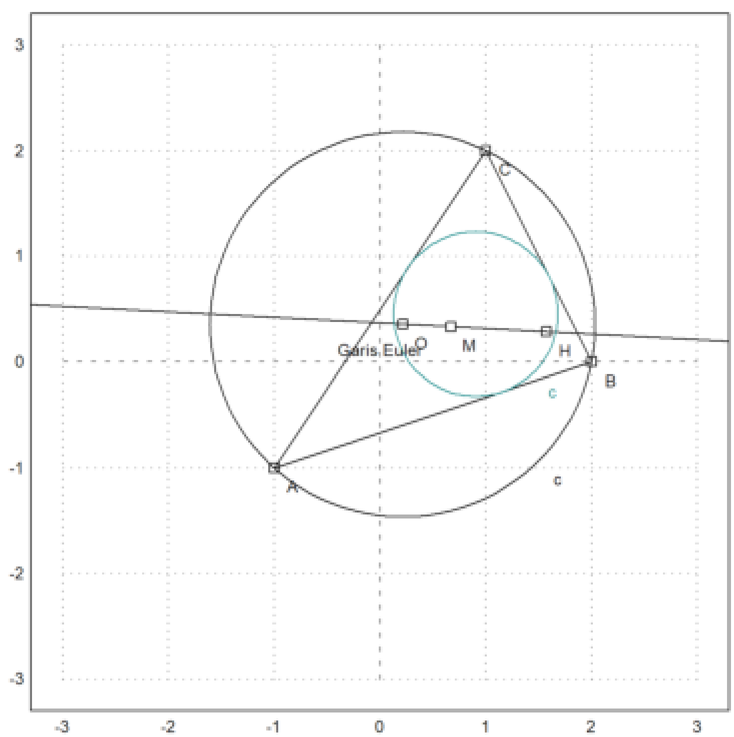
>r &= distance(Q,projectToLine(Q,lineThrough(A,B))) | ratsimp; $r

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

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

Mari kita tambahkan ini ke dalam plot.

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



Gambar 452: images/GEOMETRI_Diva%20Nagita_(23030630024)-105.png

Parabola

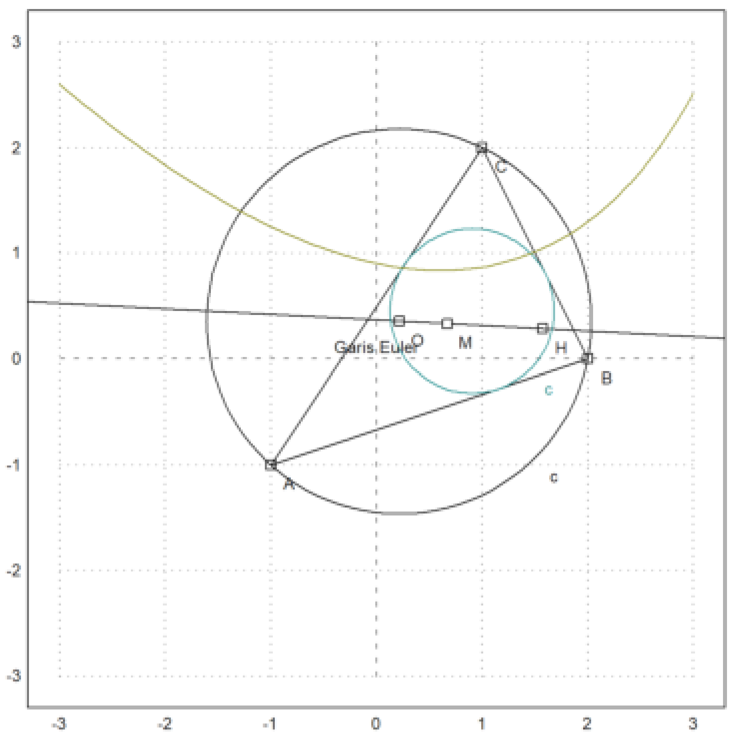
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

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

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

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

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



Gambar 453: images/GEOMETRI_Diva%20Nagita_(23030630024)-107.png

This should be some function, but the default solver of Maxima can find the solution only, if we square the equation. Consequently, we get a fake solution.

>akar &= solve(getHesseForm(lineThrough(A,B),x,y,C)$^{\text{2-distance([x,y],C)}}$2,y)

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

Solusi pertama adalah

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

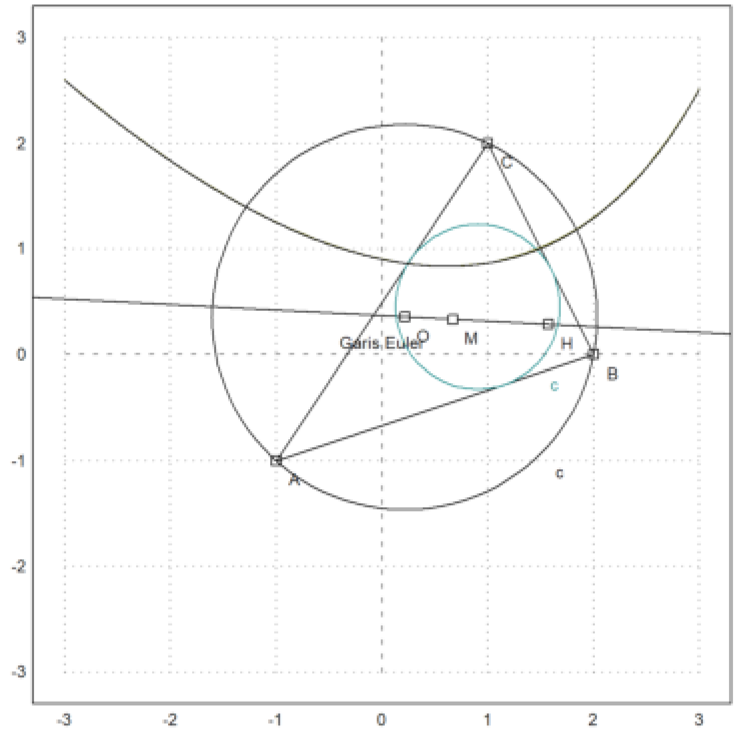
Menambahkan solusi pertama ke dalam plot menunjukkan, bahwa ini memang jalur yang kita cari. Teori mengatakan bahwa ini adalah sebuah parabola yang diputar.

>plot2d(&rhs(akar[1]),add=1):

>function g(x) &= rhs(akar[1]); $'g(x)= g(x)// fungsi yang mendefinisikan kurva di atas

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

>T &=[-1, g(-1)]; // ambil sebarang titik pada kurva tersebut



Gambar 454: images/GEOMETRI_Diva%20Nagita_(23030630024)-109.png

>dTC &= distance(T,C); $fullratsimp(dTC), $float(%) // jarak T ke C

$$\sqrt{1503 - 54\sqrt{11}\sqrt{70}}$$

$$2.135605779339061$$

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

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

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

$$\sqrt{1503 - 54\sqrt{11}\sqrt{70}}$$

$$2.135605779339061$$

Ternyata jarak T ke C sama dengan jarak T ke AB. Coba Anda pilih titik T yang lain dan ulangi perhitungan-perhitungan di atas untuk menunjukkan bahwa hasilnya juga sama.

Trigonometri Rasional

Ini terinspirasi dari sebuah ceramah N.J. Wildberger. Dalam bukunya "Proporsi Ilahi", Wildberger mengusulkan untuk mengganti gagasan klasik tentang jarak dan sudut dengan kuadransi dan penyebaran. Dengan menggunakan ini, memang memungkinkan untuk menghindari fungsi trigonometri dalam banyak contoh, dan tetap "rasional".

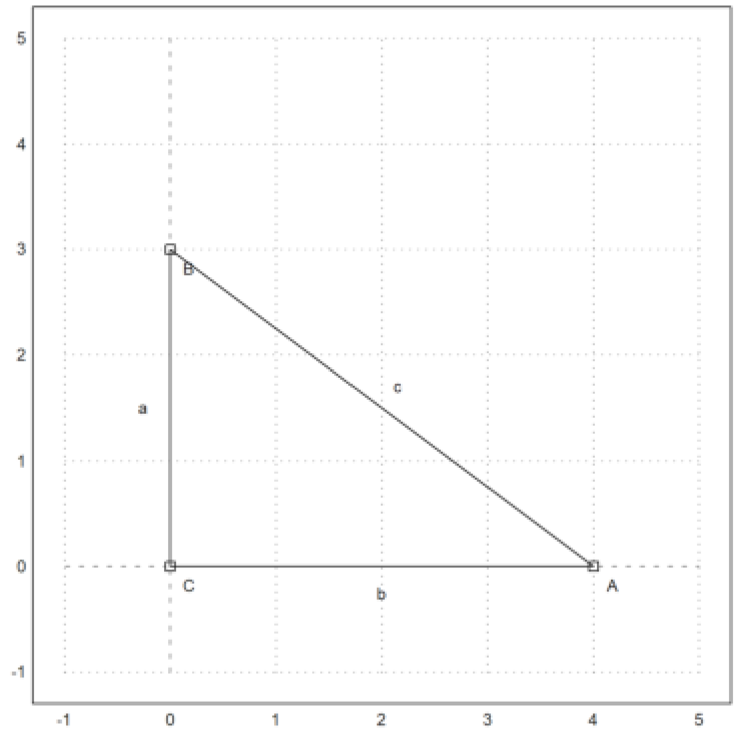
Berikut ini, saya akan memperkenalkan konsep-konsep tersebut, dan memecahkan beberapa masalah. Saya menggunakan komputasi simbolik Maxima di sini, yang menyembunyikan keuntungan utama dari trigonometri rasional yaitu komputasi dapat dilakukan dengan kertas dan pensil saja. Anda dipersilakan untuk memeriksa hasilnya tanpa komputer.

Intinya adalah bahwa komputasi rasional simbolik sering kali memberikan hasil yang sederhana. Sebaliknya, trigonometri klasik menghasilkan hasil trigonometri yang rumit, yang dievaluasi dengan pendekatan numerik saja.

>load geometry;

Untuk pengenalan pertama, kita menggunakan segitiga persegi panjang dengan proporsi Mesir yang terkenal 3, 4, dan 5. Perintah berikut ini adalah perintah Euler untuk memplot geometri bidang yang terdapat pada 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);



Gambar 455: images/GEOMETRI_Diva%20Nagita_(23030630024)-116.png

Tentu saja,

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

di mana wa adalah sudut di 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 trigonometri rasional adalah kuadrat, yang menggantikan jarak. Sebenarnya, ini hanyalah jarak yang dikuadratkan. Berikut ini, a, b, dan c menunjukkan kuadran sisi-sisinya.

Teorema Pythogoras menjadi a + b = c.

>a &= 3ˆ2; b &= 4ˆ2; c &= 5ˆ2; &a+b=c

```
25 = 25
```

Gagasan kedua dari trigonometri rasional adalah penyebaran. Penyebaran mengukur bukaan di antara garis-garis. Ini adalah 0, jika garis-garisnya sejajar, dan 1, jika garis-garisnya persegi panjang. Ini adalah kuadrat dari sinus sudut antara dua garis.

Penyebaran garis AB dan AC pada gambar di atas didefinisikan sebagai

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

di mana a dan c adalah kuadran dari segitiga persegi panjang dengan satu sudut di A.

>sa &= a/c; $sa

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

Tentu saja, hal ini lebih mudah dihitung daripada sudut. Tetapi Anda kehilangan sifat bahwa sudut dapat ditambahkan dengan mudah.

Tentu saja, kita bisa mengonversi nilai perkiraan kita untuk sudut wa ke sprad, dan mencetaknya sebagai pecahan.

>fracprint(sin(wa)^2)

```
9/25
```

Hukum kosinus trgonometri klasik diterjemahkan ke dalam “hukum silang” berikut ini.

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

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

Hukum-hukum ini diimplementasikan dalam file geometri.e yang kita masukkan ke dalam Euler.

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

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

Dalam kasus kami, kami mendapatkan

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

$$1024=1024$$

Mari kita gunakan crosslaw ini untuk mencari sebaran di A. Untuk melakukannya, kita buat crosslaw untuk kuadran a, b, dan c, dan selesaikan untuk sebaran sa yang tidak diketahui.

Anda bisa melakukan ini dengan tangan dengan mudah, tapi saya menggunakan Maxima. Tentu saja, kita mendapatkan hasil yang sudah kita dapatkan.

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

$$1024=1600\,\left(1-x\right)$$

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

Kita sudah mengetahui hal ini. Definisi penyebaran adalah kasus khusus dari crosslaw.

Kita juga dapat menyelesaikannya untuk a, b, c secara umum. Hasilnya adalah sebuah rumus yang menghitung penyebaran sudut sebuah segitiga dengan kuadran ketiga sisinya.

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

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

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

>$spread(a,b,c)

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

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

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

Hasilnya adalah rasional, yang tidak mudah didapat jika kita menggunakan trigonometri klasik.

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

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

Ini adalah sudut dalam derajat.

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

```
67°47'32.44''
```

Contoh Lain

Sekarang, mari kita coba contoh yang lebih lanjut.

Kita tentukan 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 dari file Euler untuk geometri. Jarak fungsi menggunakan geometri klasik.

>$distance(A,B)

$$\sqrt{10}$$

Euler juga memiliki fungsi untuk kuadranan antara dua titik.

Pada contoh berikut, karena c+b bukan a, maka segitiga tersebut tidak berbentuk persegi panjang.

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

$$10$$

$$5$$

$$17$$

Pertama, mari kita menghitung sudut tradisional. Fungsi computeAngle menggunakan metode yang biasa berdasarkan hasil kali titik dari dua vektor. Hasilnya adalah beberapa perkiraan titik mengambang.

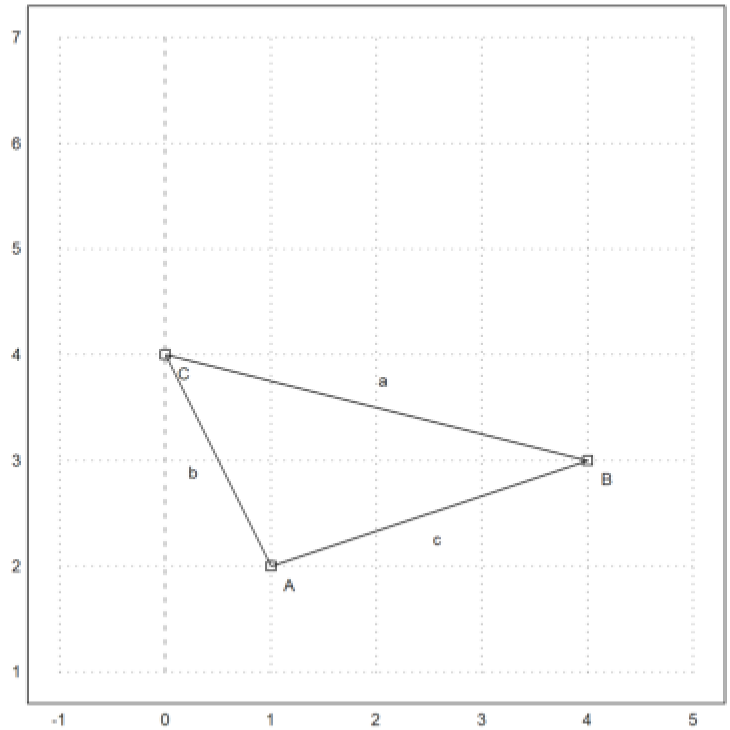
$$A =< 1, 2 > \quad B =< 4, 3 >, \quad C =< 0, 4 >$$

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

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

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

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



Gambar 456: images/GEOMETRI_Diva%20Nagita_(23030630024)-129.png

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

32.4711922908

Dengan menggunakan pensil dan kertas, kita dapat melakukan hal yang sama dengan hukum silang. Kita masukkan kuadran a, b, dan c ke dalam hukum silang dan selesaikan untuk x.

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

$$4 = 200\,(1 - x)$$

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

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

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

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

Maxima mendapatkan hasil yang sama dengan menggunakan trigonometri biasa, jika kita memaksakannya. Ia menyelesaikan suku sin(arccos(. . .)) menjadi hasil pecahan. Sebagian besar siswa tidak dapat melakukan ini.

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

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

Setelah kita memiliki penyebaran di B, kita dapat menghitung tinggi ha di sisi a. Ingatlah bahwa

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

menurut definisi.

>ha &= c*sb; $ha

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

Gambar berikut ini dibuat dengan program geometri C.a.R., yang dapat menggambar kuadran dan penyebaran.

image: (20) Rational_Geometry_CaR.png

Menurut definisi, panjang ha adalah akar kuadrat dari kuadrannya.

>$sqrt(ha)

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

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

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

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

Rumus penentu yang biasa menghasilkan hasil yang sama.

>$areaTriangle(B,A,C)

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

Rumus Heron

Sekarang, mari kita selesaikan masalah ini secara umum!

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

Pertama-tama kita menghitung penyebaran di B untuk segitiga dengan sisi a, b, dan c. Kemudian kita menghitung luas kuadrat ("quadrea"?), memfaktorkannya dengan Maxima, dan kita mendapatkan rumus Heron yang terkenal.

Memang, hal ini sulit dilakukan dengan pensil dan kertas.

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

$$\frac{-c^4-\left(-2\,b^2-2\,a^2\right)\,c^2-b^4+2\,a^2\,b^2-a^4}{4\,a^2\,c^2}$$
$$\frac{\left(-c+b+a\right)\,\left(c-b+a\right)\,\left(c+b-a\right)\,\left(c+b+a\right)}{16}$$

Aturan Triple Spread

Kerugian dari spread adalah bahwa mereka tidak lagi hanya menambahkan sudut seperti.

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

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

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

Aturan ini berlaku untuk tiga sudut yang berjumlah 180°.

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

Karena spread dari

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

adalah sama, aturan triple spread juga benar, jika

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

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

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

Sebagai contoh, kita dapat menghitung penyebaran sudut 60°. Ini adalah 3/4. Namun, persamaan ini memiliki solusi kedua, di mana semua penyebarannya adalah 0.

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

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

Penyebaran 90° jelas adalah 1. Jika dua sudut ditambahkan ke 90°, penyebarannya akan menyelesaikan persamaan penyebaran tiga dengan a, b, 1. Dengan perhitungan berikut, kita mendapatkan a + b = 1.

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

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

$$[x = 1 - y]$$

Karena penyebaran 180°-t sama dengan penyebaran t, rumus penyebaran tiga kali lipat juga berlaku, jika satu sudut adalah jumlah atau selisih dari dua sudut lainnya.

Jadi kita dapat menemukan penyebaran sudut dua kali lipat. Perhatikan bahwa ada dua solusi lagi. Kita jadikan ini sebuah fungsi.

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

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

- 4 (a - 1) a

Pembagi Sudut

Ini adalah situasi yang sudah kita ketahui.

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

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

>&remvalue(a,b,c);

Jadi, pertama-tama kita menghitung penyebaran sudut yang dibelah dua di A, menggunakan rumus penyebaran tiga.

Masalah dengan rumus ini muncul lagi. Rumus ini memiliki dua solusi. Kita harus memilih yang benar. Solusi lainnya mengacu pada sudut terbagi dua 180°-wa.

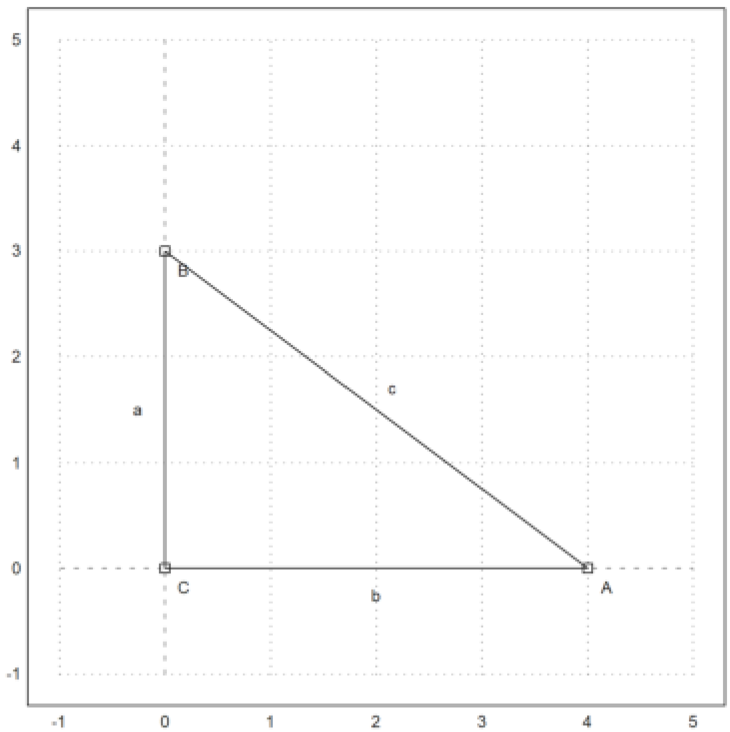
>$triplespread(x,x,a/(a+b)), $solve(%,x), sa2 &= rhs(%[1]); $sa2

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

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

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

Mari kita periksa persegi panjang Mesir.



Gambar 457: images/GEOMETRI_Diva%20Nagita_(23030630024)-159.png

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

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

Kita bisa mencetak sudut dalam Euler, setelah mentransfer penyebaran ke radian.
>wa2 := arcsin(sqrt(1/10)); degprint(wa2)

```
18°26'5.82''
```

Titik P adalah perpotongan garis bagi sudut dengan sumbu y.
>P := [0,tan(wa2)*4]

```
[0,  1.33333]
```

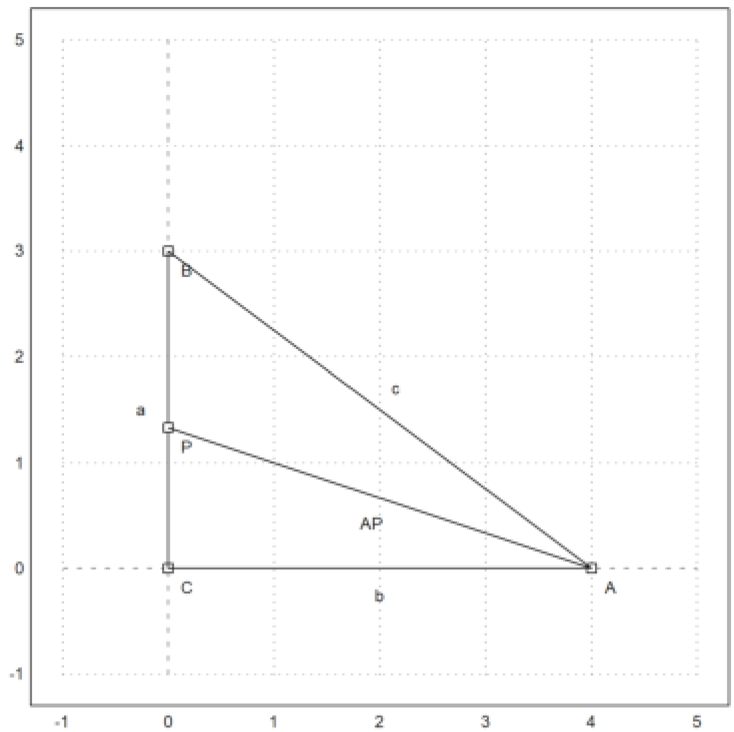
>plotPoint(P,"P"); plotSegment(A,P):
Mari kita periksa sudut-sudutnya dalam contoh spesifik kita.
>computeAngle(C,A,P), computeAngle(P,A,B)

```
0.321750554397
0.321750554397
```

Sekarang kita hitung panjang garis bagi AP.
Kita menggunakan teorema sinus dalam segitiga APC. Teorema ini menyatakan bahwa

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

berlaku untuk semua segitiga. Kuadratkan, ini diterjemahkan ke dalam apa yang disebut "hukum penyebar-an"



Gambar 458: images/GEOMETRI_Diva%20Nagita_(23030630024)-164.png

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

di mana a, b, c menunjukkan qudrance.

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

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

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

Mari kita periksa rumus ini untuk nilai-nilai Mesir kita.

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

```
4.21637021356
4.21637021356
```

Kita juga dapat menghitung P dengan menggunakan rumus penyebaran.

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

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

Nilainya sama dengan yang kita dapatkan dengan rumus trigonometri.

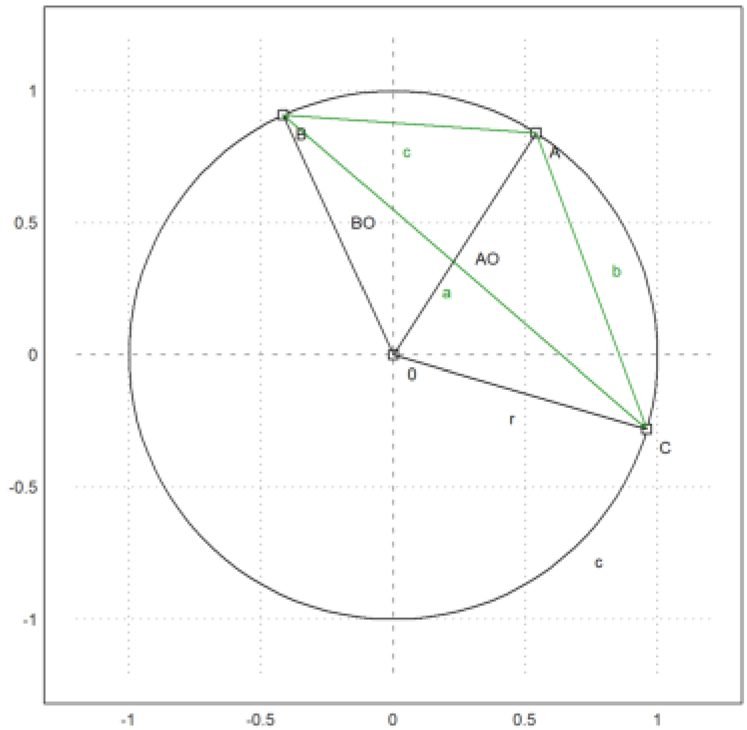
>sqrt(mxmeval("at(py,[a=3$^{2,b=4}$2])"))

```
1.33333333333
```

Sudut Akor

Lihatlah situasi berikut ini.

>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;



Gambar 459: images/GEOMETRI_Diva%20Nagita_(23030630024)-169.png

Kita dapat menggunakan Maxima untuk menyelesaikan rumus penyebaran tiga untuk sudut-sudut di pusat O untuk r. Dengan demikian kita mendapatkan rumus untuk jari-jari kuadrat dari pericircle dalam hal kuadran sisi-sisinya.

Kali ini, Maxima menghasilkan beberapa angka nol yang rumit, yang kita abaikan.

>&remvalue(a,b,c,r); // hapus nilai-nilai sebelumnya untuk perhitungan baru

>rabc &= rhs(solve(triplespread(spread(b,r,r),spread(a,r,r),spread(c,r,r)),r)[4]); $rabc

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

Kita dapat menjadikannya sebuah fungsi Euler.

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

Mari kita periksa hasilnya untuk poin A, B, C.

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

Radiusnya memang 1.

>periradius(a,b,c)

1

Faktanya adalah, bahwa penyebaran CBA hanya bergantung pada b dan c. Ini adalah teorema sudut akor.

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

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

Faktanya, penyebarannya adalah b/(4r), dan kita melihat bahwa sudut chord b adalah setengah dari sudut tengah.

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

$$0$$

Contoh 6: Jarak Minimal pada Bidang

Keterangan Awal

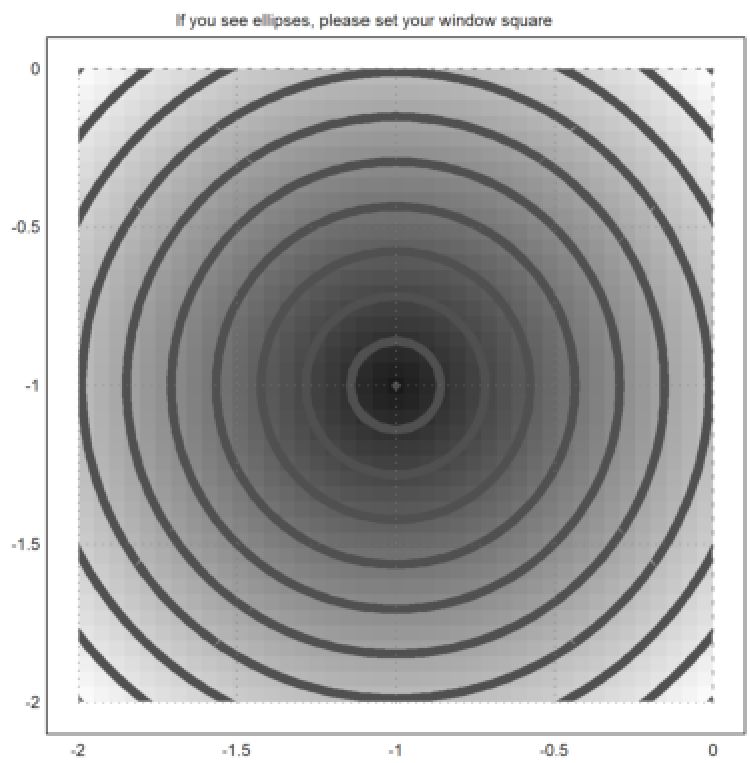
Fungsi yang, pada sebuah titik M pada bidang, menetapkan jarak AM antara titik tetap A dan M, memiliki garis-garis tingkat yang cukup sederhana: lingkaran yang berpusat di A.

>&remvalue();

>A=[-1,-1];

>function d1(x,y):=sqrt((x-A[1])$^{2+(y-A[2])}$2)

>fcontour("d1",xmin=-2,xmax=0,ymin=-2,ymax=0,hue=1, ...
> title="If you see ellipses, please set your window square"):

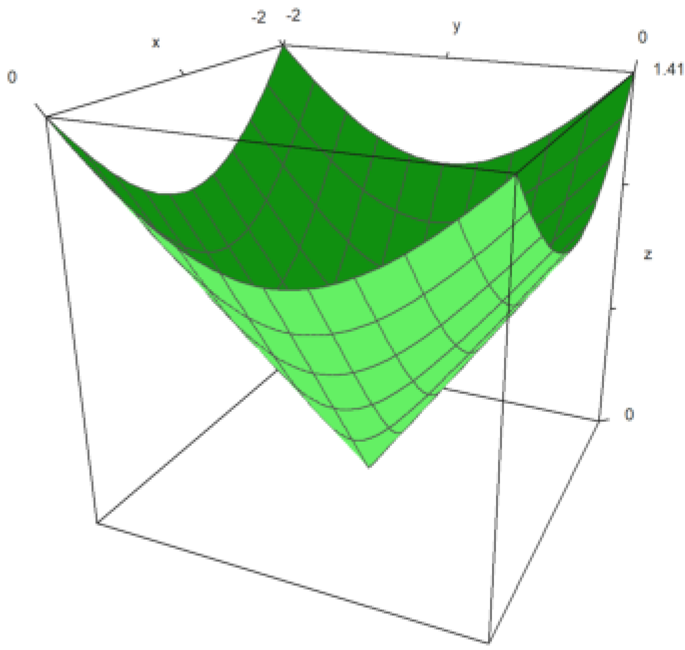


Gambar 460: images/GEOMETRI_Diva%20Nagita_(23030630024)-173.png

dan grafiknya juga cukup sederhana: bagian atas kerucut:

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

Tentu saja nilai minimum 0 diperoleh dalam A.



Gambar 461: images/GEOMETRI_Diva%20Nagita_(23030630024)-174.png

Dua Titik

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

>B=[1,-1];

>function d2(x,y):=d1(x,y)+sqrt((x-B[1])$^{2+(y-B[2])}$2)

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

Grafiknya lebih menarik:

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

Pembatasan pada garis (AB) lebih terkenal:

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

Tiga Poin

Sekarang, hal-hal menjadi kurang sederhana: Hal ini sedikit kurang dikenal bahwa MA+MB+MC mencapai minimumnya pada satu titik di bidang, tetapi untuk menentukannya tidak sesederhana itu:

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

Contoh:

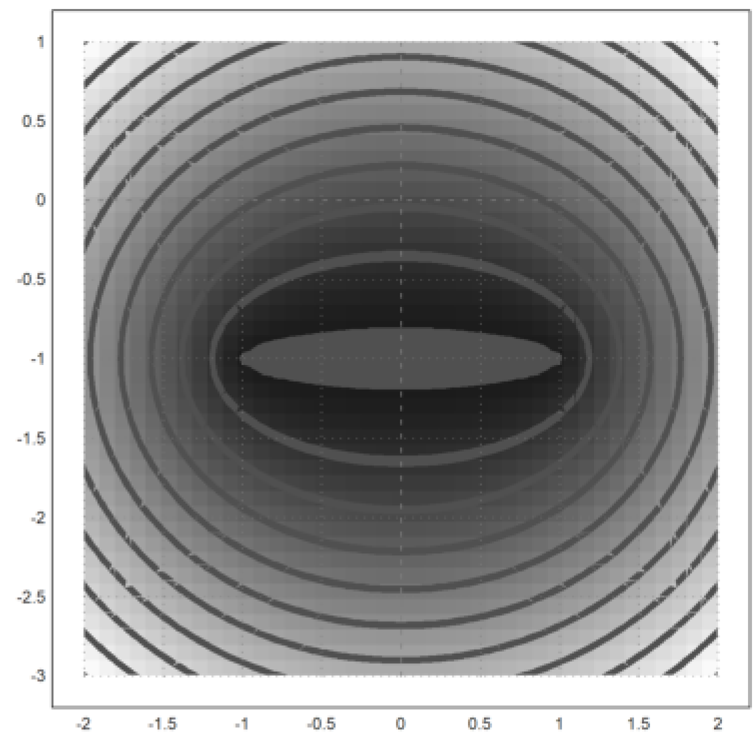
>C=[-4,1];

>function d3(x,y):=d2(x,y)+sqrt((x-C[1])$^{2+(y-C[2])}$2)

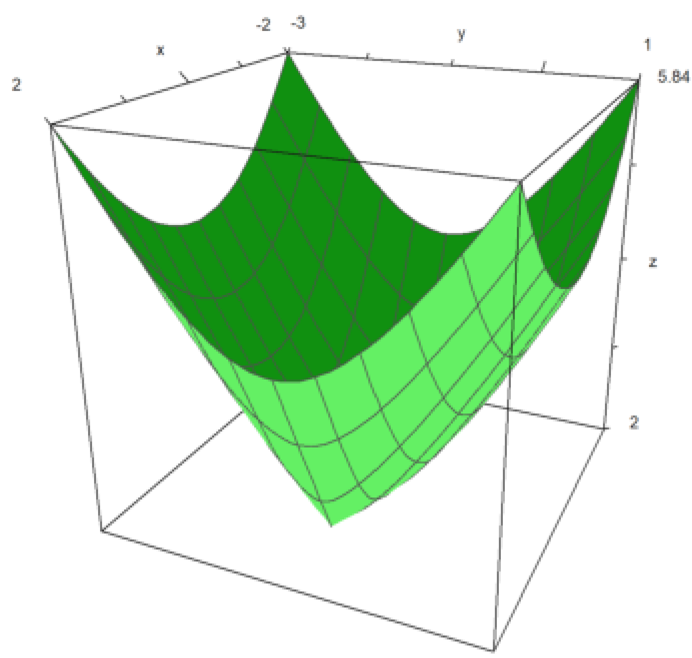
>plot3d("d3",xmin=-5,xmax=3,ymin=-4,ymax=4);

>insimg;

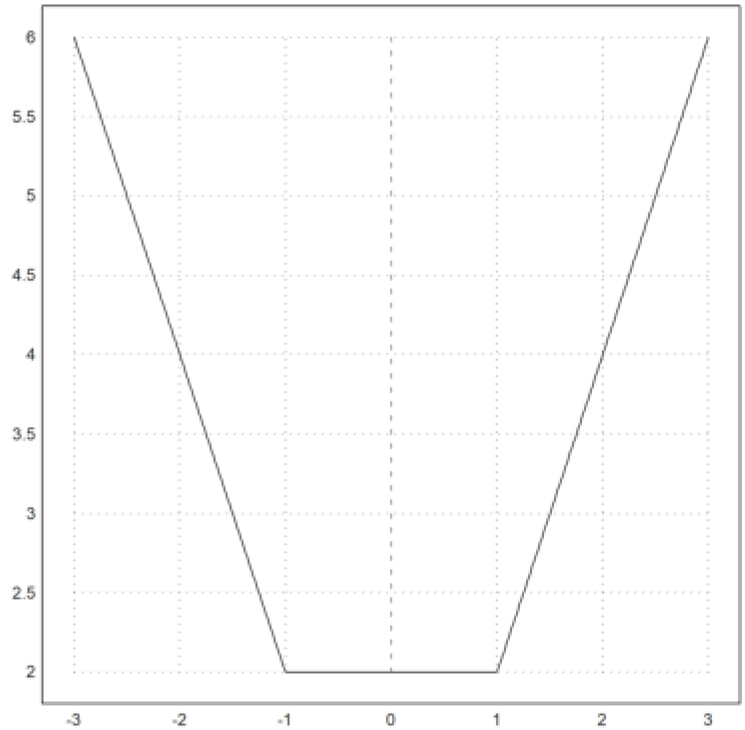
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A");



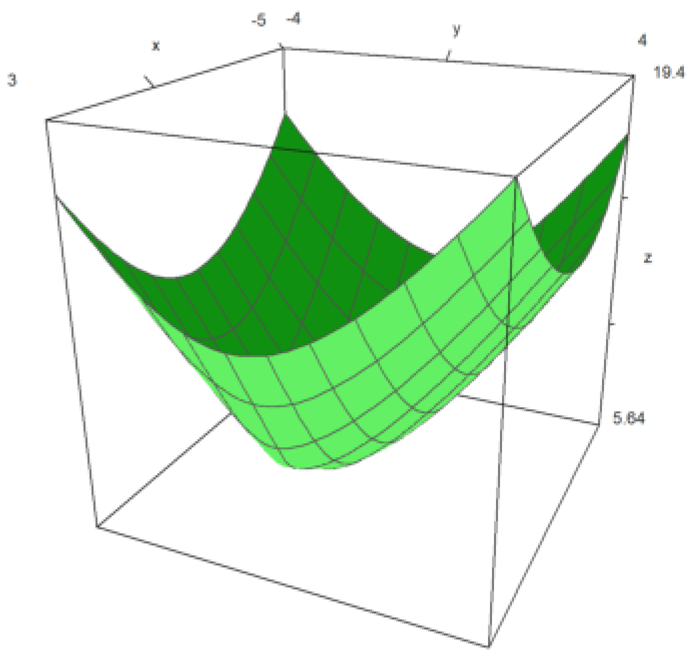
Gambar 462: images/GEOMETRI_Diva%20Nagita_(23030630024)-175.png



Gambar 463: images/GEOMETRI_Diva%20Nagita_(23030630024)-176.png



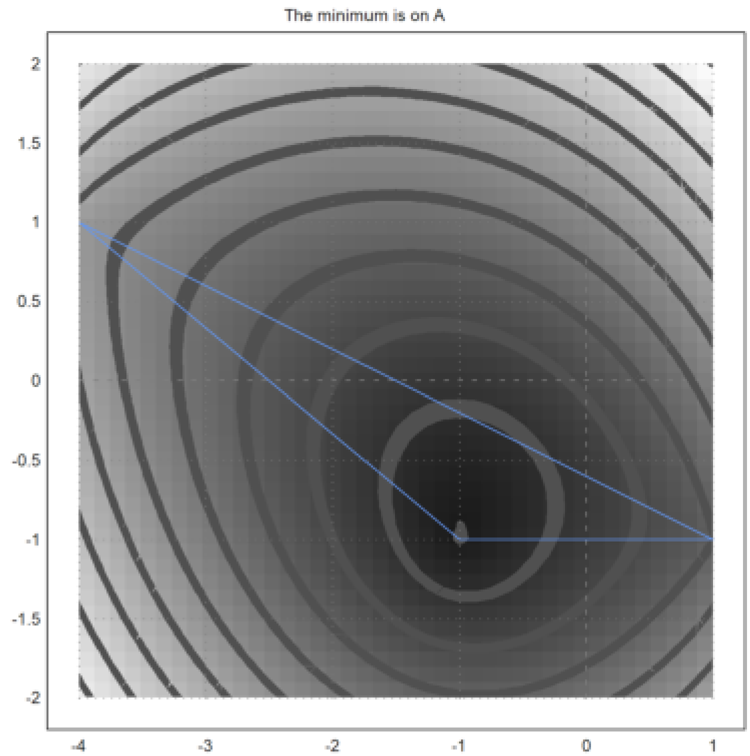
Gambar 464: images/GEOMETRI_Diva%20Nagita_(23030630024)-177.png



Gambar 465: images/GEOMETRI_Diva%20Nagita_(23030630024)-178.png

>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);

>insimg;



Gambar 466: images/GEOMETRI_Diva%20Nagita_(23030630024)-179.png

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

>C=[-0.5,1];

>plot3d("d3",xmin=-2,xmax=2,ymin=-2,ymax=2):

>fcontour("d3",xmin=-2,xmax=2,ymin=-2,ymax=2,hue=1,title="The Fermat point");

>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);

>insimg;

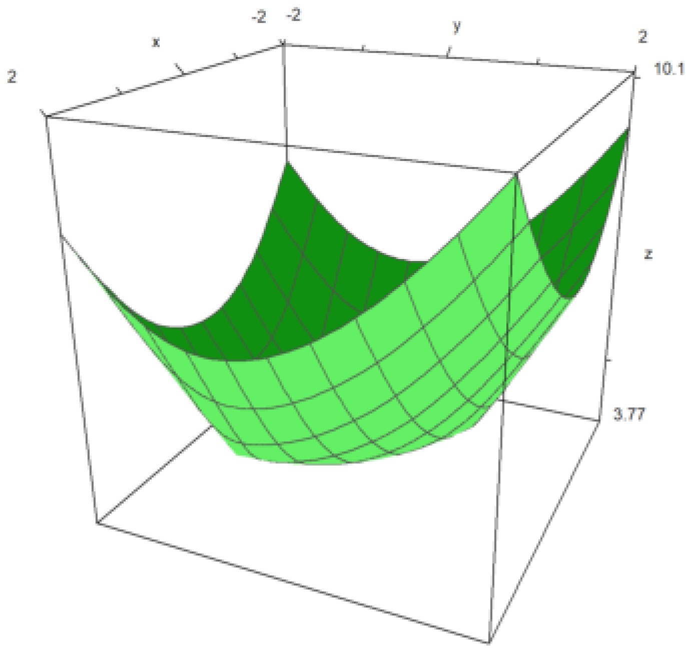
Merupakan kegiatan yang menarik untuk merealisasikan gambar di atas dengan perangkat lunak geometri; sebagai contoh, saya tahu sebuah perangkat lunak yang ditulis dalam bahasa Java yang memiliki instruksi "garis kontur"...

Semua hal di atas telah ditemukan oleh seorang hakim Perancis bernama Pierre de Fermat; dia menulis surat kepada para ahli lain seperti pendeta Marin Mersenne dan Blaise Pascal yang bekerja di bagian pajak penghasilan. Jadi titik unik F sedemikian rupa sehingga FA+FB+FC minimal, disebut titik Fermat dari segitiga. Namun tampaknya beberapa tahun sebelumnya, Torriccelli dari Italia telah menemukan titik ini sebelum Fermat menemukannya! Bagaimanapun juga, tradisinya adalah mencatat titik F ini...

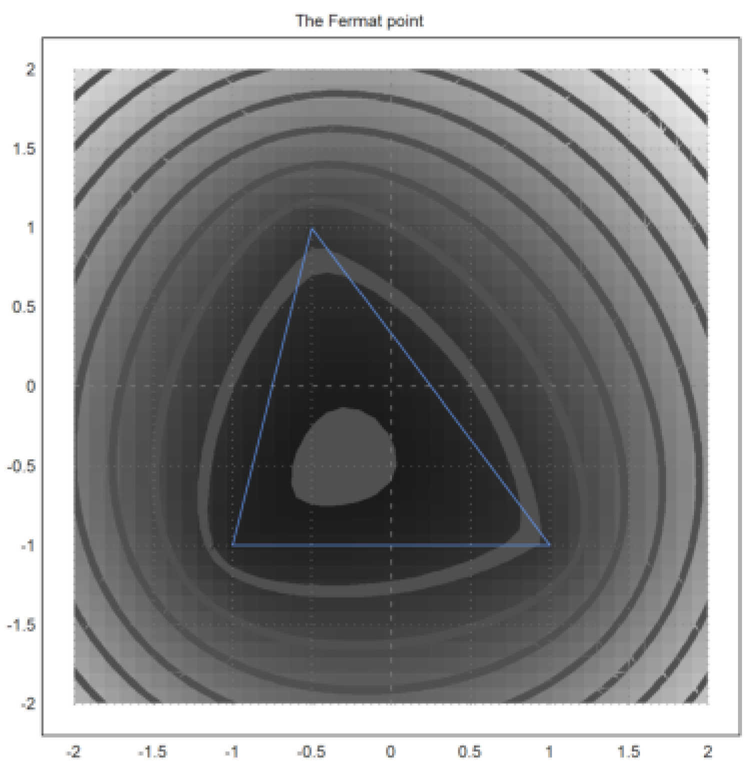
Empat Poin

Langkah selanjutnya adalah menambahkan titik ke-4 D dan mencoba meminimumkan MA+MB+MC+MD; misalkan Anda adalah operator TV kabel dan ingin menemukan di bidang mana Anda harus meletakkan antena sehingga Anda dapat memberi makan empat desa dan menggunakan panjang kabel sesedikit mungkin!

>D=[1,1];

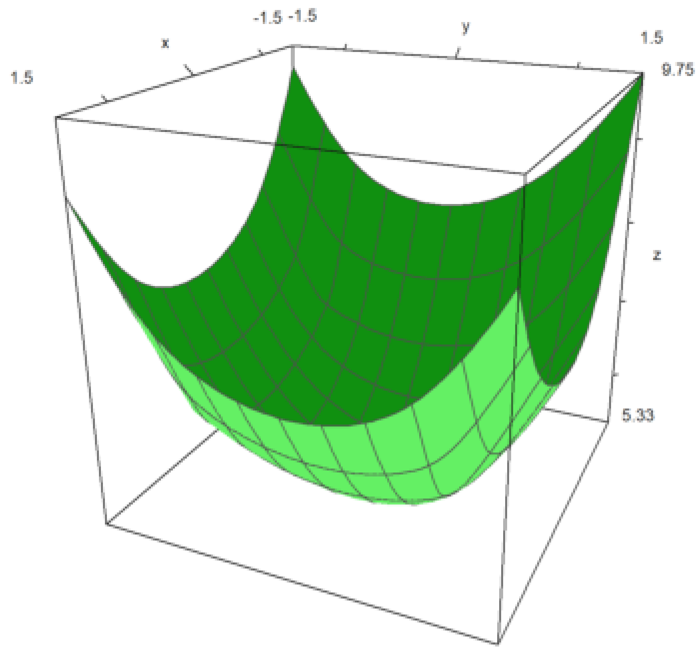


Gambar 467: images/GEOMETRI_Diva%20Nagita_(23030630024)-180.png



Gambar 468: images/GEOMETRI_Diva%20Nagita_(23030630024)-181.png

>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):



Gambar 469: images/GEOMETRI_Diva%20Nagita_(23030630024)-182.png

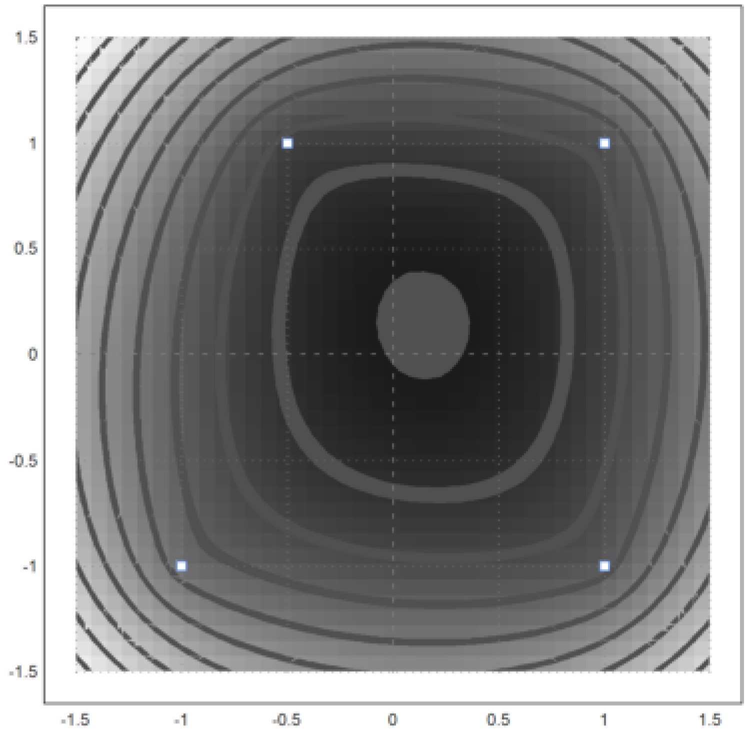
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
>P=(A_B_C_D)'; plot2d(P[1],P[2],points=1,add=1,color=12);
>insimg;
Masih ada nilai minimum dan tidak ada simpul A, B, C, maupun D:
>function f(x):=d4(x[1],x[2])
>neldermin("f",[0.2,0.2])

```
[0.142858,  0.142857]
```

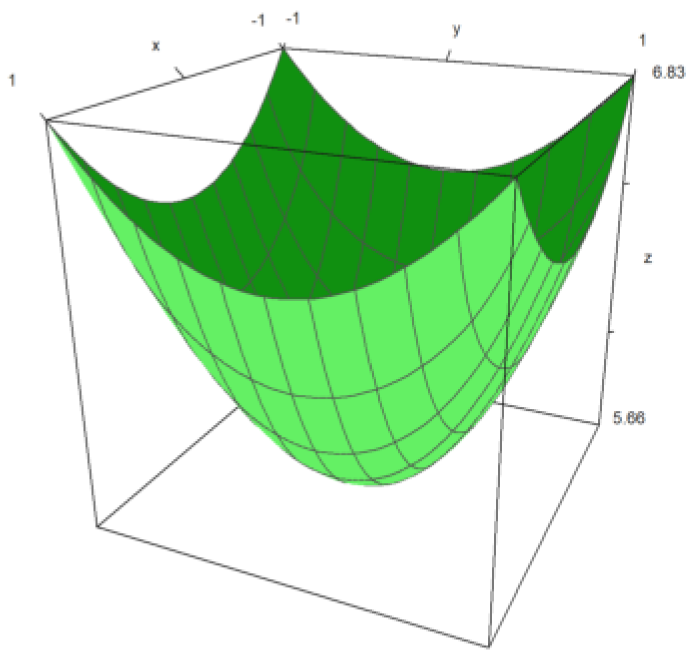
Tampaknya dalam kasus ini, koordinat titik optimal adalah rasional atau mendekati rasional. . .
Karena ABCD adalah sebuah bujur sangkar, kita berharap bahwa titik optimalnya adalah pusat dari ABCD:
>C=[-1,1];
>plot3d("d4",xmin=-1,xmax=1,ymin=-1,ymax=1):
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
>P=(A_B_C_D)'; plot2d(P[1],P[2],add=1,color=12,points=1);
>insimg;

Bola Dandelin dengan Povray

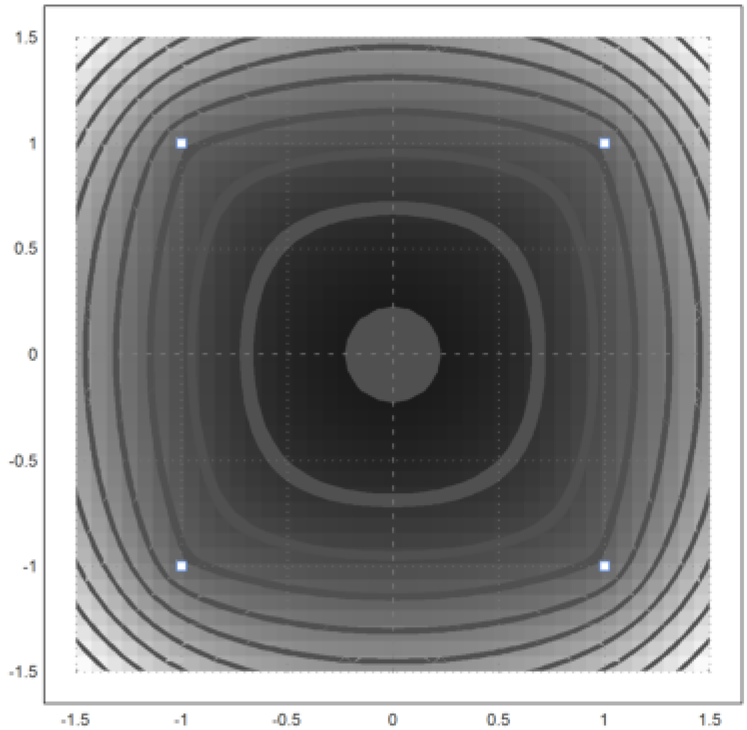
Anda dapat menjalankan demonstrasi ini, jika Anda telah menginstal Povray, dan pvengine.exe pada path program.
Pertama, kita menghitung jari-jari bola.



Gambar 470: images/GEOMETRI_Diva%20Nagita_(23030630024)-183.png



Gambar 471: images/GEOMETRI_Diva%20Nagita_(23030630024)-184.png



Gambar 472: images/GEOMETRI_Diva%20Nagita_(23030630024)-185.png

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

Kita menggunakan file geometri.e dari Euler untuk hal ini.

>load geometry;

Pertama, dua garis yang membentuk kerucut.

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

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

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

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

Kemudian baris ketiga.

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

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

Kami merencanakan semuanya sejauh ini.

>setPlotRange(-1,1,0,2);

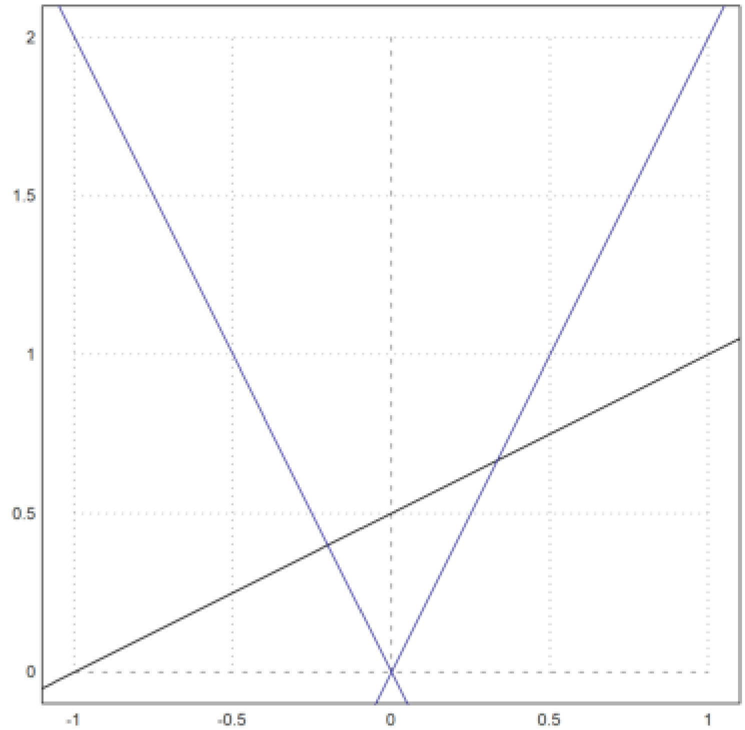
>color(black); plotLine(g(),“ “)

>a:=2; color(blue); plotLine(g1(),“ “), plotLine(g2(),” “):

Sekarang, kita ambil titik umum pada sumbu y.

>P &= [0,u]

```
[0, u]
```



Gambar 473: images/GEOMETRI_Diva%20Nagita_(23030630024)-186.png

Hitung jarak ke g1.
>d1 &= distance(P,projectToLine(P,g1)); $d1

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

Hitung jarak ke g.
>d &= distance(P,projectToLine(P,g)); $d

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

Dan temukan pusat kedua lingkaran, yang jaraknya sama.
>sol &= solve(d1$^{2=d}$2,u); $sol

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

Ada dua solusi.
Kami mengevaluasi solusi simbolis, dan menemukan kedua pusat, dan kedua jarak.
>u := sol()

```
[0.333333,  1]
```

>dd := d()

```
[0.149071,  0.447214]
```

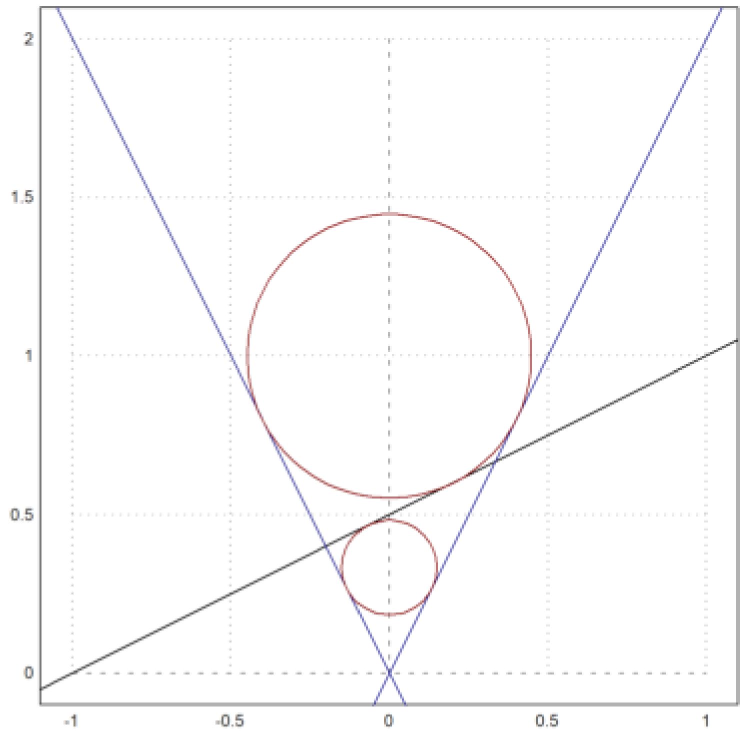
Plot lingkaran-lingkaran tersebut ke dalam gambar.

>color(red);

>plotCircle(circleWithCenter([0,u[1]],dd[1]),“ ”);

>plotCircle(circleWithCenter([0,u[2]],dd[2]),“ ”);

>insimg;



Gambar 474: images/GEOMETRI_Diva%20Nagita_(23030630024)-190.png

Plot dengan Povray

Selanjutnya kita plot semuanya dengan Povray. Perhatikan bahwa Anda mengubah perintah apapun pada urutan perintah Povray berikut ini, dan jalankan kembali semua perintah dengan Shift-Return.

Pertama 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 pemandangan dengan tepat.

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

Selanjutnya kita tulis kedua bola tersebut ke 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)));

Kami menghasilkan bidang yang terbatas 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 menghasilkan dua titik pada lingkaran, di mana bola menyentuh kerucut.

>function turnz(v) := return

>P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]);

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

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

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

Kemudian, kita menghasilkan dua titik di mana bola-bola tersebut menyentuh bidang. 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)));

Selanjutnya kita menghitung perpotongan P1P2 dengan bidang.

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

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

Kami 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 menghasilkan pita abu-abu, di mana bola-bola menyentuh kerucut.

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

>pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2],1);

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

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

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

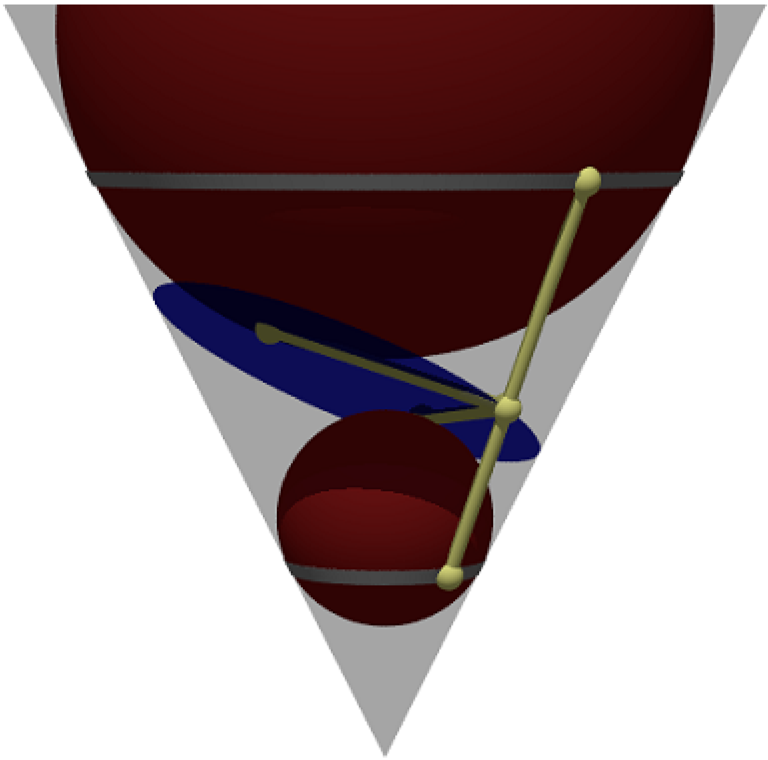
Mulai program Povray.

>povend();

Untuk mendapatkan Anaglyph ini, kita harus 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)));
```



Gambar 475: images/GEOMETRI_Diva%20Nagita_(23030630024)-191.png

```
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/sian untuk mengapresiasi efek berikut ini.

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

Geometri Bumi

Pada buku catatan ini, kita ingin melakukan beberapa komputasi bola. Fungsi-fungsi tersebut terdapat pada file "spherical.e" pada folder contoh. Kita perlu memuat file tersebut terlebih dahulu.

>load "spherical.e";

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

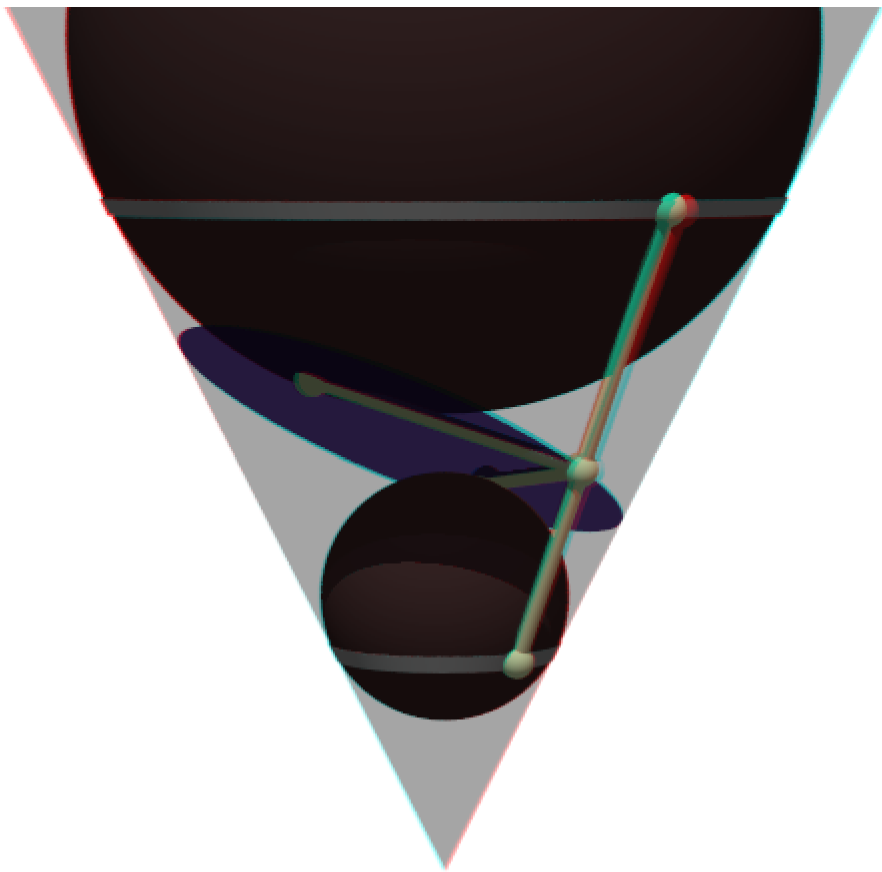
>FMIPA=[rad(-7,-46.467),rad(110,23.05)]

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

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

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

```
S 7°46.467' E 110°23.050'
```



Gambar 476: images/GEOMETRI_Diva%20Nagita_(23030630024)-192.png

Mari kita tambahkan dua kota lagi, Solo dan Semarang.

>Solo=[rad(-7,-34.333),rad(110,49.683)]; Semarang=[rad(-6,-59.05),rad(110,24.533)];

>sposprint(Solo), sposprint(Semarang),

```
S 7°34.333' E 110°49.683'
S 6°59.050' E 110°24.533'
```

Pertama, kita menghitung vektor dari satu titik ke titik lainnya pada bola ideal. Vektor ini adalah [heading, 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''
```

Ini adalah perkiraan yang baik. Rutinitas berikut ini menggunakan perkiraan yang lebih baik lagi. Pada jarak yang begitu dekat, hasilnya hampir sama.

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

Ada fungsi untuk judul, dengan mempertimbangkan bentuk elips bumi. Sekali lagi, kami mencetak dengan cara yang canggih.

>sdegprint(esdir(FMIPA,Solo))

```
     65.34°
```

Sudut segitiga melebihi 180° pada bola.

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

```
180°0'10.77''
```

Ini bisa digunakan untuk menghitung luas segitiga. Catatan: Untuk segitiga kecil, cara ini tidak akurat karena kesalahan pengurangan dalam asum-pi.

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

Ada sebuah fungsi untuk hal ini, yang menggunakan garis lintang rata-rata segitiga untuk menghitung radius bumi, dan menangani kesalahan pembulatan untuk segitiga yang sangat kecil.

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

```
2123.64310526 km^2
```

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

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

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

Fungsi-fungsi ini mengasumsikan bola yang ideal. Hal yang sama 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 beralih ke contoh yang lebih besar, Tugu Jogja dan Monas Jakarta (menggunakan Google Earth untuk mencari koordinatnya).

>Tugu=[-7.7833°,110.3661°]; Monas=[-6.175°,106.811944°];

>sposprint(Tugu), sposprint(Monas)

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

Menurut Google Earth, jaraknya adalah 429,66 km. Kami mendapatkan perkiraan yang bagus.

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

Judulnya sama dengan yang dihitung di Google Earth.

>degprint(esdir(Tugu,Monas))

```
294°17'2.85''
```

Namun demikian, kita tidak lagi mendapatkan posisi target yang tepat, jika kita menambahkan arah dan jarak ke posisi semula. Hal ini terjadi, karena kita tidak menghitung fungsi inversi secara tepat, tetapi mengambil perkiraan radius bumi di sepanjang jalur.

>sposprint(esadd(Tugu,esdir(Tugu,Monas),esdist(Tugu,Monas)))

```
S 6°10.500' E 106°48.717'
```

Namun demikian, kesalahannya tidak besar.

>sposprint(Monas),

```
S 6°10.500' E 106°48.717'
```

Tentu saja, kita tidak bisa berlayar dengan arah yang sama dari satu tujuan ke tujuan lainnya, jika kita ingin mengambil jalur terpendek. Bayangkan, Anda terbang ke arah NE mulai dari titik mana pun di bumi. Kemudian Anda akan berputar ke kutub utara. Lingkaran besar tidak mengikuti arah yang konstan!

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

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

Sekarang kita tambahkan 10 kali sepersepuluh dari jarak tersebut, dengan menggunakan arah menuju Monas, kita akan sampai di Tugu.

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

Hasilnya jauh berbeda.

>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°];

Jalur terpendek dari P1 ke P2 bukanlah lingkaran lintang 30°, tetapi jalur 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 jarak total.

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

```
     79.69°
     81.67°
```

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

Jaraknya tidak tepat, karena kita akan menambahkan sedikit kesalahan, jika kita mengikuti judul yang sama terlalu lama.

>skmprint(esdist(p,P2))

```
0.203km
```

Kita akan mendapatkan perkiraan yang baik, jika kita menyesuaikan arah setiap 1/100 dari total jarak dari Tugu ke Monas.

>p=Tugu; dist=esdist(Tugu,Monas); . . .
> loop 1 to 100; p=esadd(p,esdir(p,Monas),dist/100); end;

>skmprint(esdist(p,Monas))

```
0.000km
```

Untuk keperluan navigasi, kita bisa mendapatkan urutan posisi GPS di sepanjang Bundaran Hotel Indonesia menuju Monas dengan fungsi navigate.

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

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

Kami menulis sebuah fungsi, yang memplot bumi, dua posisi, dan posisi di antaranya.

>function testplot . . .

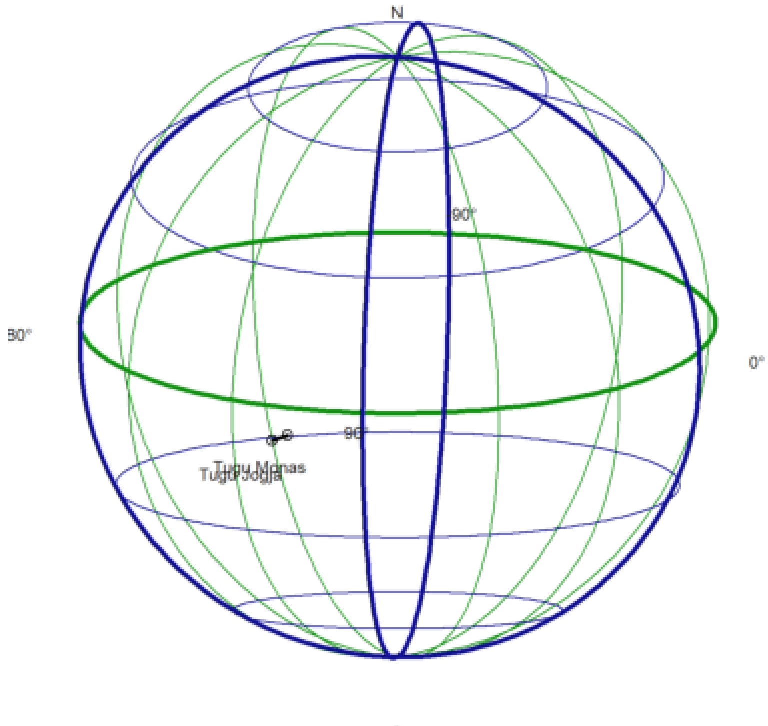
```
useglobal;
plotearth;
plotpos(Tugu,"Tugu Jogja"); plotpos(Monas,"Tugu Monas");
plotposline(v);
endfunction
```

Sekarang rencanakan semuanya.

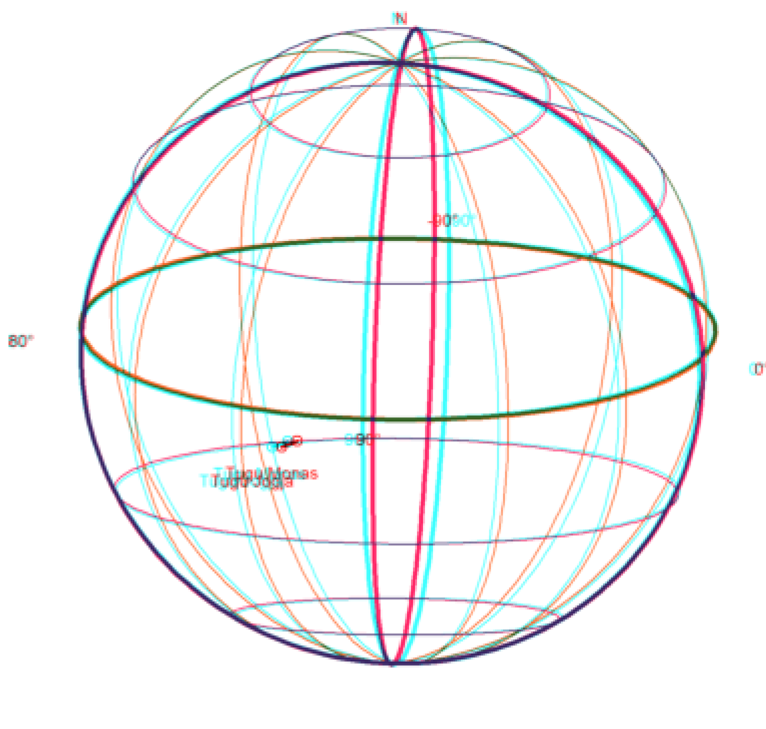
>plot3d("testplot",angle=25, height=6,>own,>user,zoom=4):

Atau gunakan plot3d untuk mendapatkan tampilan anaglyph. Ini terlihat sangat bagus dengan kacamata merah/cyan.

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



Gambar 477: images/GEOMETRI_Diva%20Nagita_(23030630024)-193.png



Gambar 478: images/GEOMETRI_Diva%20Nagita_(23030630024)-194.png

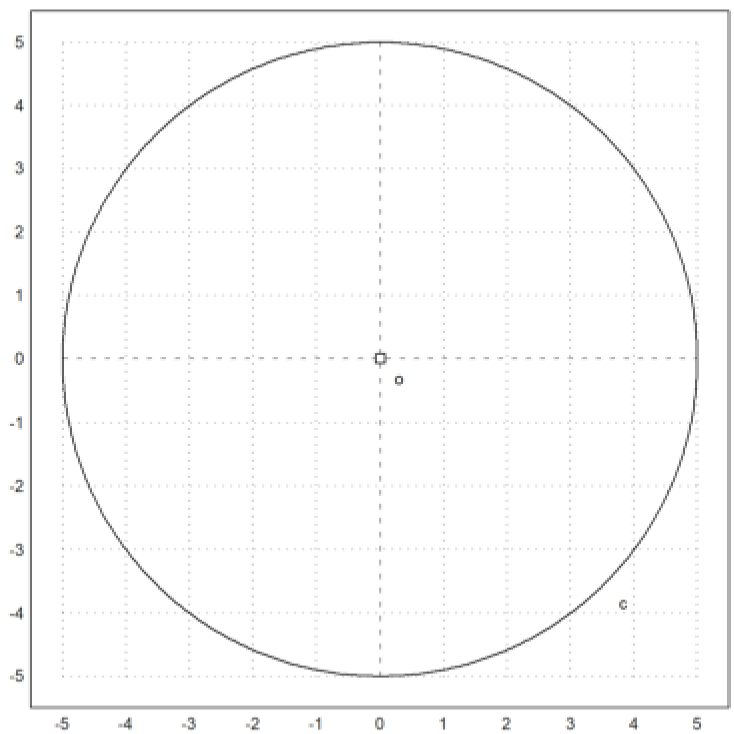
Latihan

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

Petunjuk:

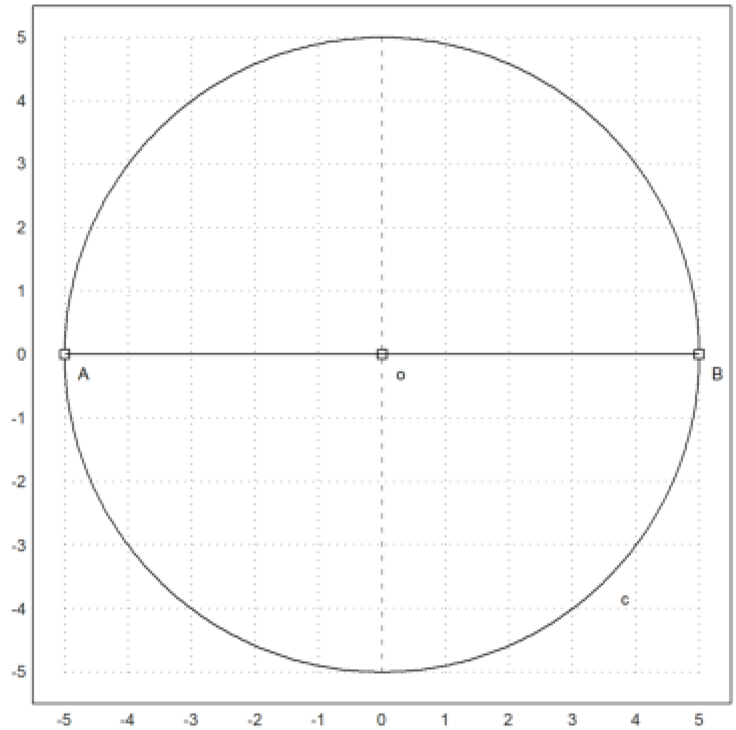
- Besar sudut pusat yang menghadap masing-masing sisi segi-n adalah
- (360/n).
- Titik-titik sudut segi-n merupakan perpotongan lingkaran luar segi-n
- dan garis-garis yang melalui pusat dan saling membentuk sudut sebesar
- kelipatan (360/n).
- Untuk n ganjil, pilih salah satu titik sudut adalah di atas.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

>o &:= [0,0]; c=circleWithCenter(o,5);
>color(1); setPlotRange(5); plotPoint(o); plotCircle(c):



Gambar 479: images/GEOMETRI_Diva%20Nagita_(23030630024)-195.png

>A=[-5,0]; plotPoint(A,"A");
>B=[5,0]; plotPoint(B,"B");
>plotSegment(A,B," "):
>c1=circleWithCenter(A,distance(A,o));



Gambar 480: images/GEOMETRI_Diva%20Nagita_(23030630024)-196.png

>c2=circleWithCenter(B,distance(B,o));
>k=circleCircleIntersections(c1,c);
>l=circleCircleIntersections(c,c2);
>m=circleCircleIntersections(c2,c);
>n=circleCircleIntersections(c,c1);
>r=lineThrough(k,m); s=lineThrough(l,n);
>setPlotRange(8); plotPoint(o); plotCircle(c); plotPoint(A,"A"); plotPoint(B,"B"); plotSegment(A,B," ");
>color(4); plotCircle(c1); plotCircle(c2); plotPoint(k); plotPoint(l); plotPoint(m); plotPoint(n)
>color(5); plotLine(r); plotLine(s):
>color(6); plotSegment(A,k," "); plotSegment(A,n,""); plotSegment(k,l,""); ...
> plotSegment(l,B,""); plotSegment(B,m,""); plotSegment(m,n," "):
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"):

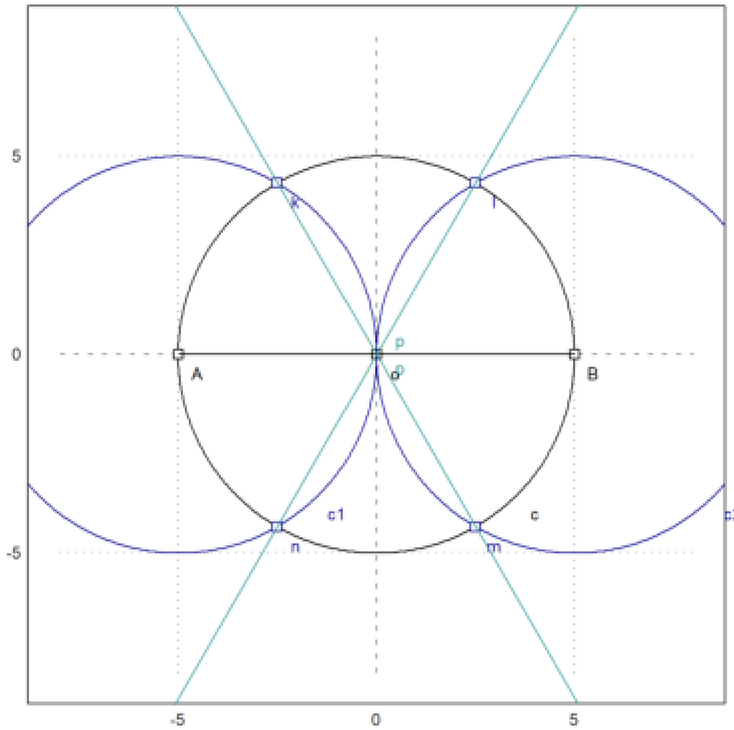
2. Gambarlah suatu parabola yang melalui 3 titik yang diketahui.

Petunjuk:

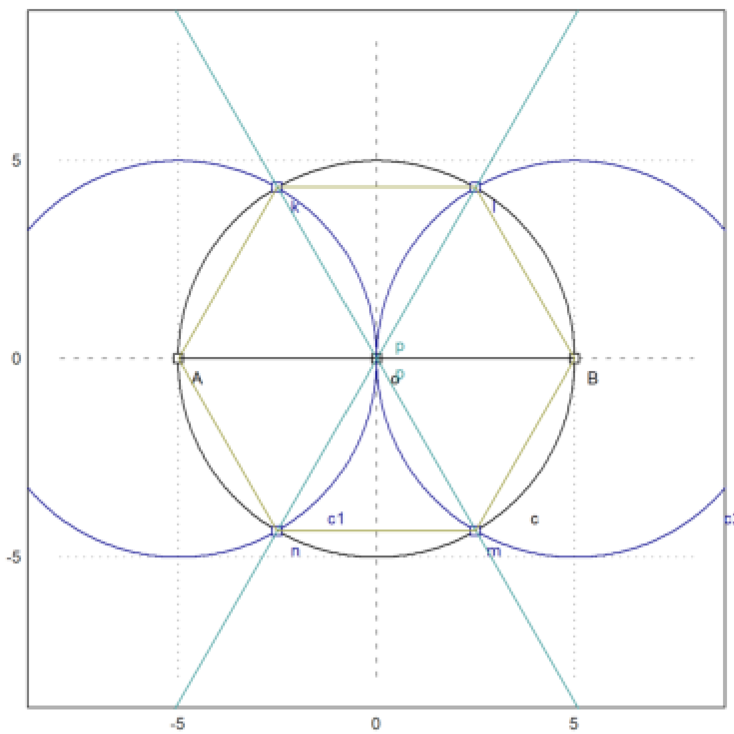
- Misalkan persamaan parabolanya y= ax^2+bx+c.
- Substitusikan koordinat titik-titik yang diketahui ke persamaan tersebut.
- Selesaikan SPL yang terbentuk untuk mendapatkan nilai-nilai a, b, c.

>setPlotRange(5); A=[1,0]; B=[4,0]; C=[0,-4];
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"):
>sol &= solve([a+b=-c,16*a+4*b=-c,c=-4],[a,b,c])

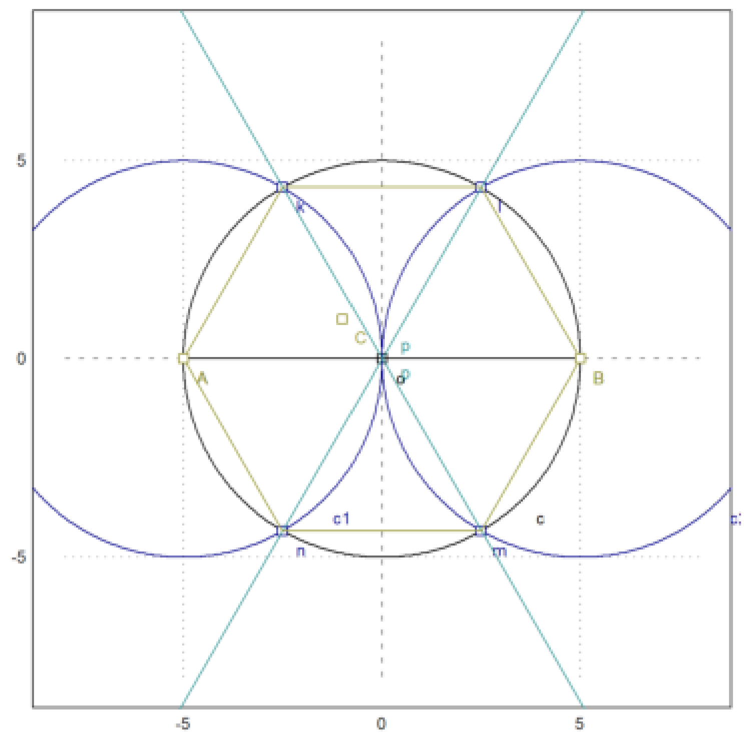
```
[[a = - 1, b = 5, c = - 4]]
```



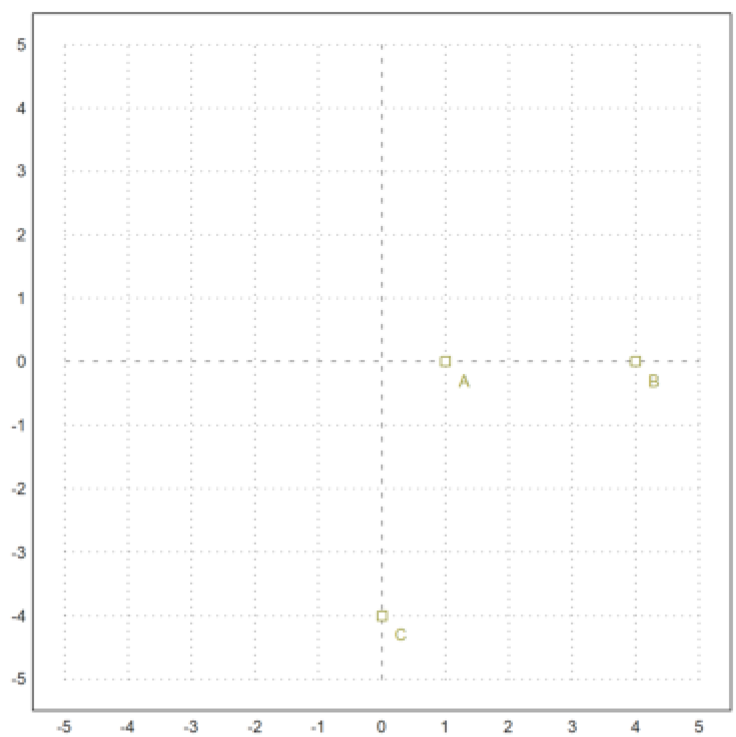
Gambar 481: images/GEOMETRI_Diva%20Nagita_(23030630024)-197.png



Gambar 482: images/GEOMETRI_Diva%20Nagita_(23030630024)-198.png



Gambar 483: images/GEOMETRI_Diva%20Nagita_(23030630024)-199.png

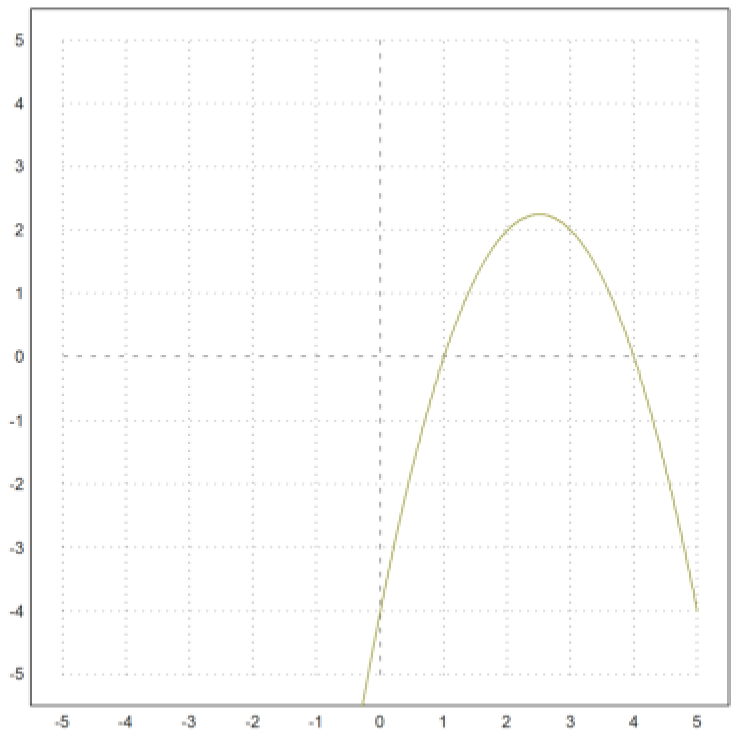


Gambar 484: images/GEOMETRI_Diva%20Nagita_(23030630024)-200.png

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

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

>plot2d("-x^2+5*x-4",-5,5,-5,5):

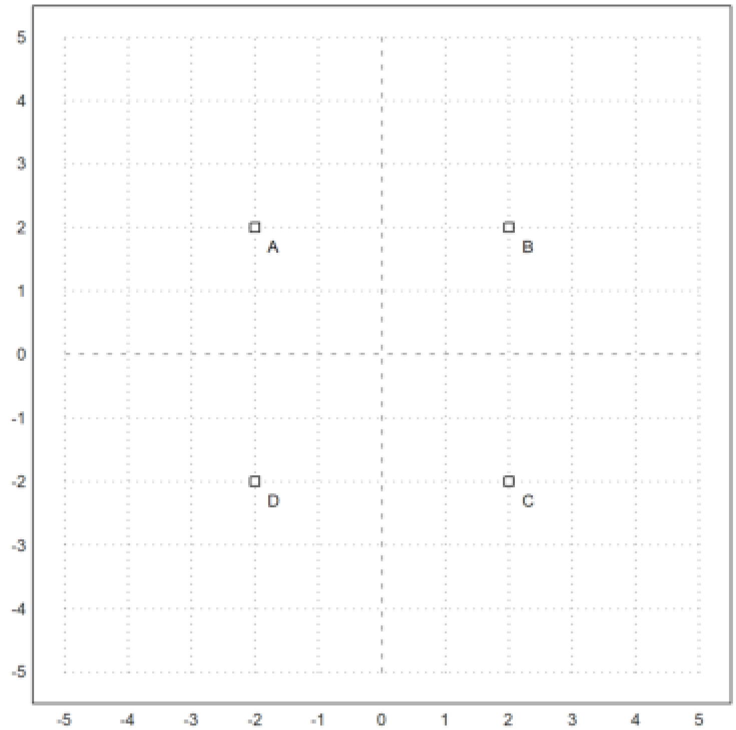


Gambar 485: images/GEOMETRI_Diva%20Nagita_(23030630024)-201.png

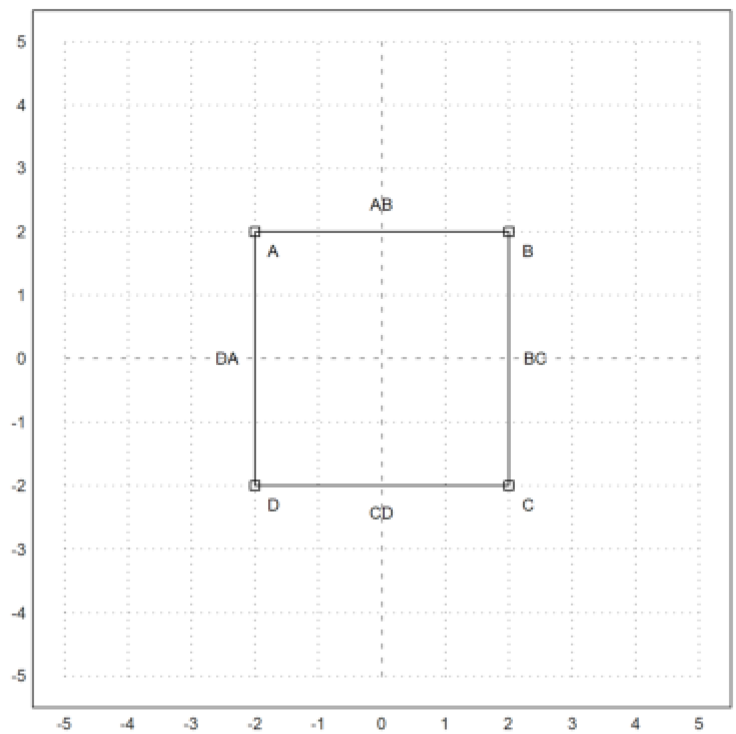
3. Gambarlah suatu segi-4 yang diketahui keempat titik sudutnya, misalnya A, B, C, D.

   - 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.

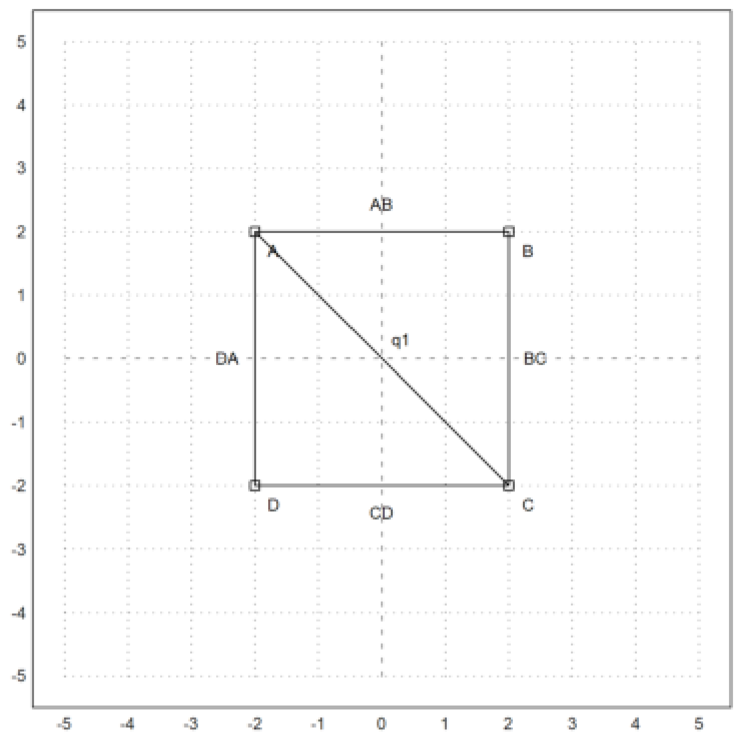
>setPlotRange(-5,5,-5,5);
>A=[-2,2]; plotPoint(A,"A");
>B=[2,2]; plotPoint(B,"B");
>C=[2,-2]; plotPoint(C,"C");
>D=[-2,-2]; plotPoint(D,"D"):
>plotSegment(A,B);
>plotSegment(B,C);
>plotSegment(C,D);
>plotSegment(D,A):
>plotSegment(A,C,"q1"):



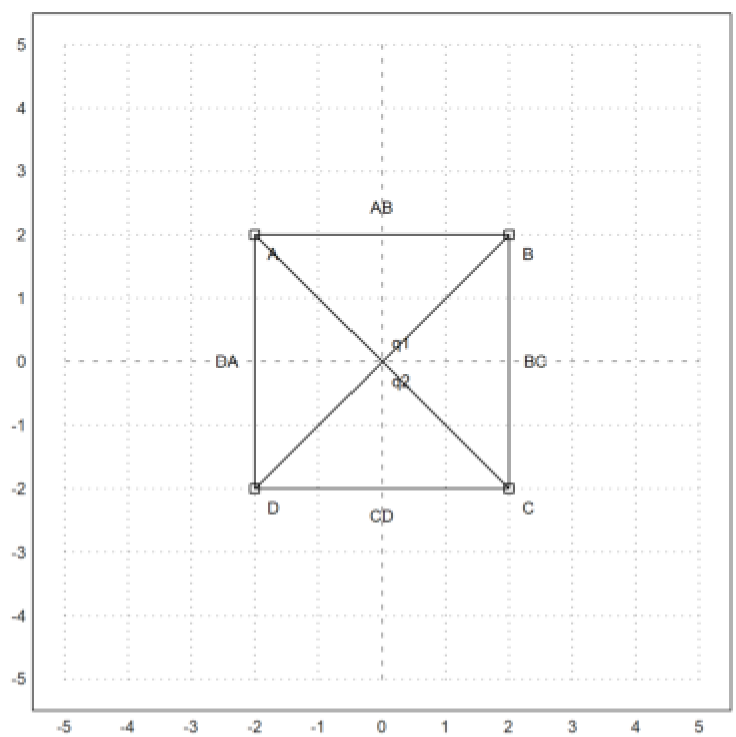
Gambar 486: images/GEOMETRI_Diva%20Nagita_(23030630024)-202.png



Gambar 487: images/GEOMETRI_Diva%20Nagita_(23030630024)-203.png

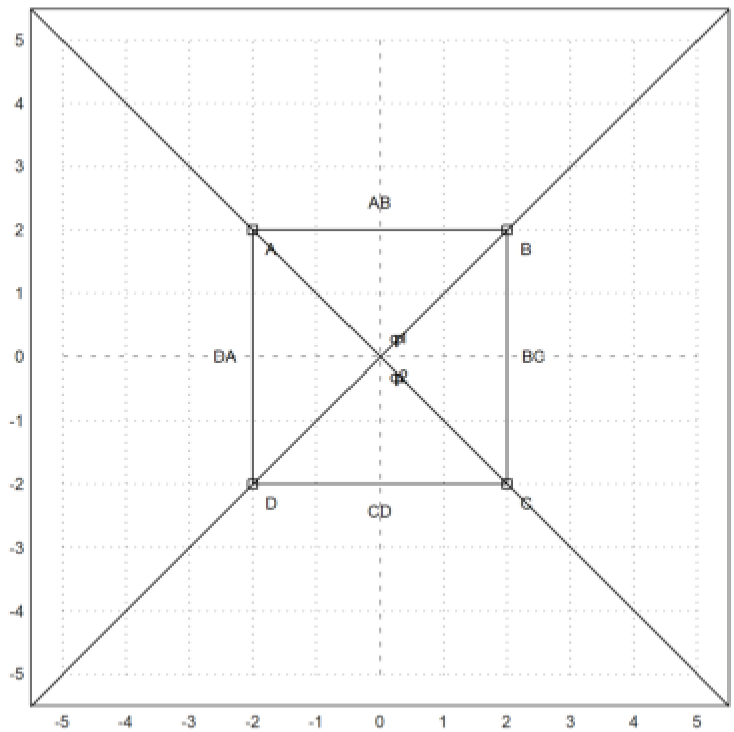


Gambar 488: images/GEOMETRI_Diva%20Nagita_(23030630024)-204.png



Gambar 489: images/GEOMETRI_Diva%20Nagita_(23030630024)-205.png

>plotSegment(B,D,“q2”):
>q1=lineThrough(A,C);
>q2=lineThrough(B,D);
>p=lineIntersection(q1,q2);
>plotLine(q1); plotLine(q2):



Gambar 490: images/GEOMETRI_Diva%20Nagita_(23030630024)-206.png

>plotPoint(p, “P”);
>r=norm(p-projectToLine(p,lineThrough(A,B)))

2

>plotCircle(circleWithCenter(p,r),“lingkaran dalam segi-4 ABCD”):
>AB=norm(A-B) // panjang sisi AB

4

>CD=norm(C-D) // panjang sisi CD

4

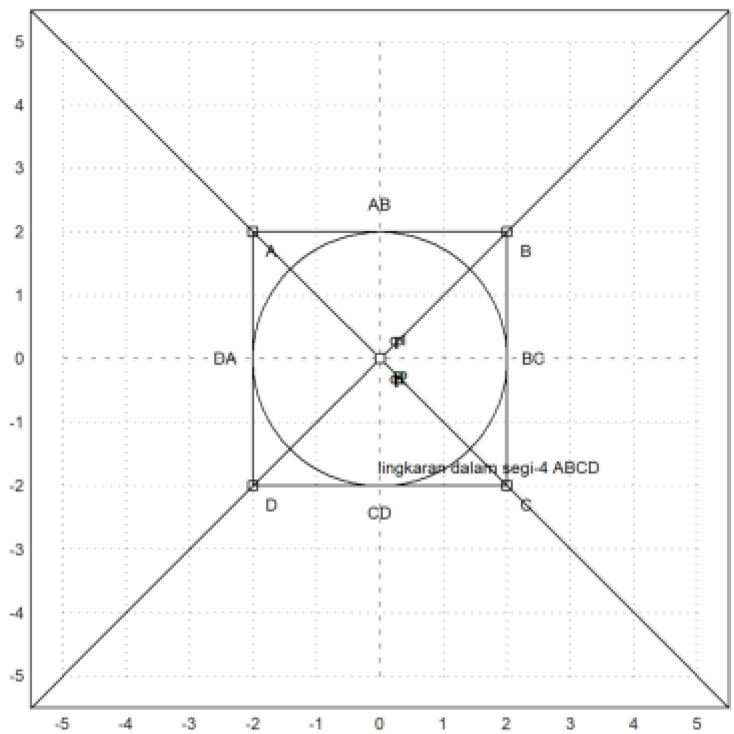
>AD=norm(A-D) // panjang sisi AD

4

>BC=norm(B-C) // panjang sisi BC

4

>AB.CD



Gambar 491: images/GEOMETRI_Diva%20Nagita_(23030630024)-207.png

```
16
```

>AD.BC

```
16
```

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

>P=[-1,-1]; Q=[1,-1];

>function d1(x,y):=sqrt((x-P[1])$^{2+(y-P[2])}$2)

>Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])$^{2+(y-P[2])}$2)+sqrt((x-Q[1])$^{2+(y-Q[2])}$2)

>fcontour(“d2”,xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):

>reset

```
0
```

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

>reset

```
0
```

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

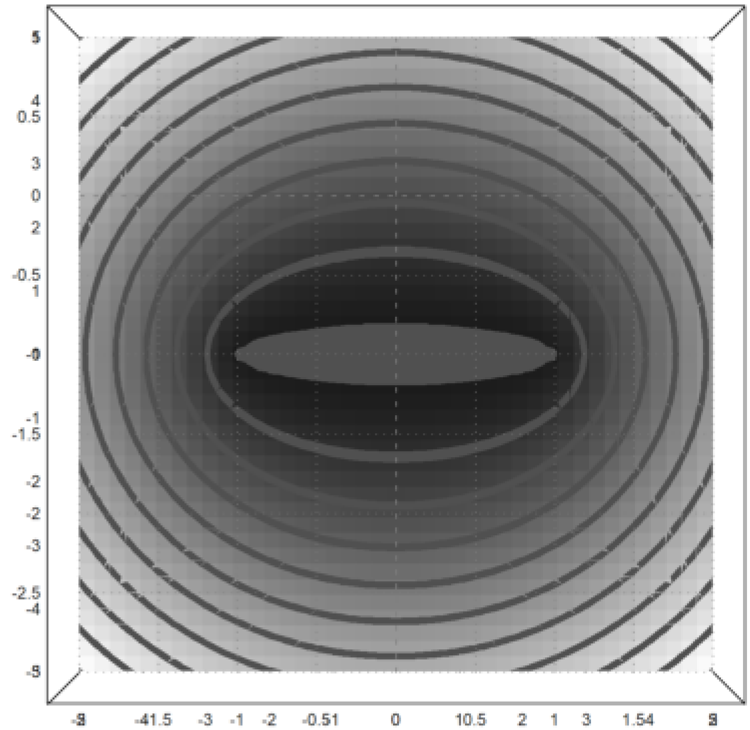
>P=[-1,-1]; Q=[1,-1];

>function d1(x,y):=sqrt((x-p[1])$^{2+(y-p[2])}$2)

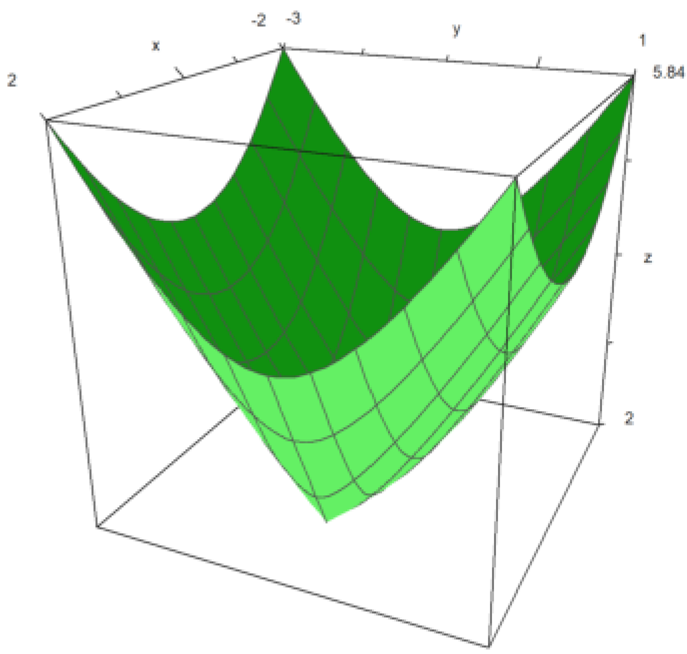
>Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])$^{2+(y-P[2])}$2)+sqrt((x+Q[1])$^{2+(y+Q[2])}$2)

>fcontour(“d2”,xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):

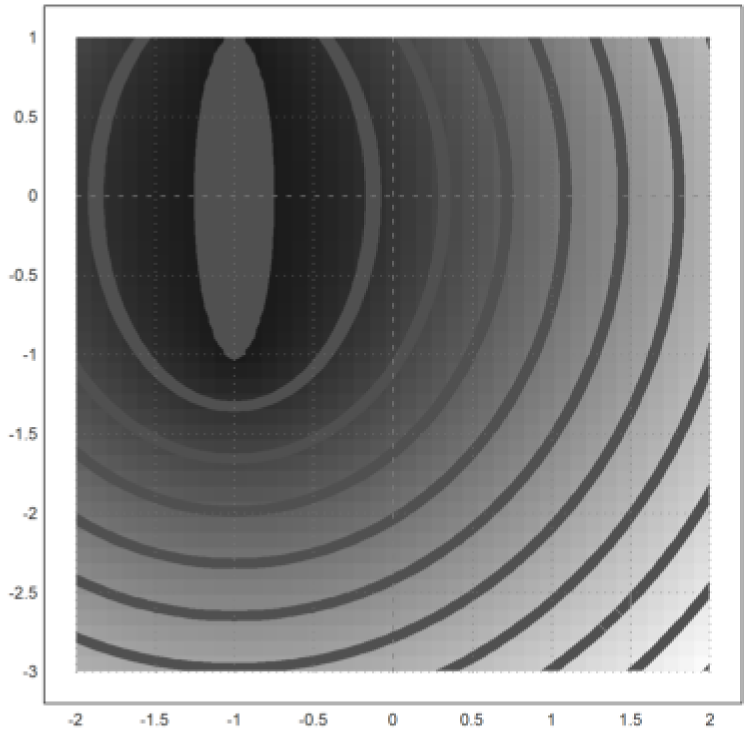
>reset



Gambar 492: images/GEOMETRI_Diva%20Nagita_(23030630024)-208.png



Gambar 493: images/GEOMETRI_Diva%20Nagita_(23030630024)-209.png



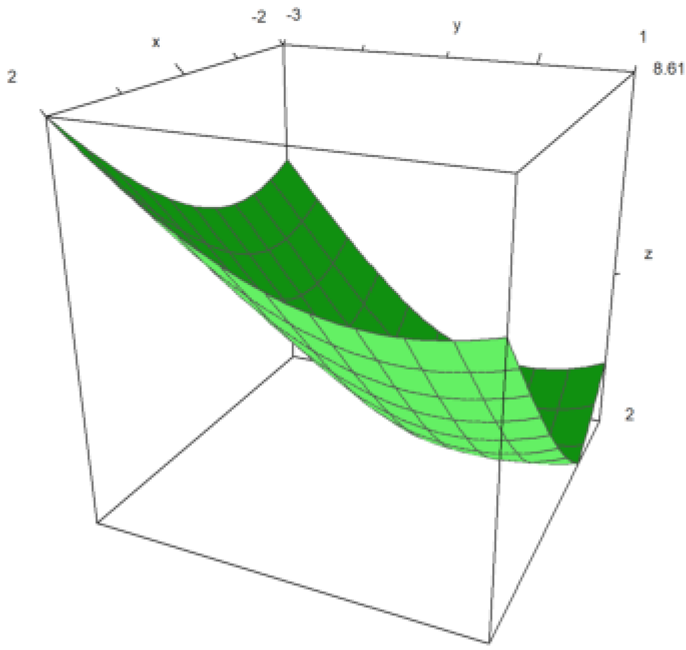
Gambar 494: images/GEOMETRI_Diva%20Nagita_(23030630024)-210.png

0

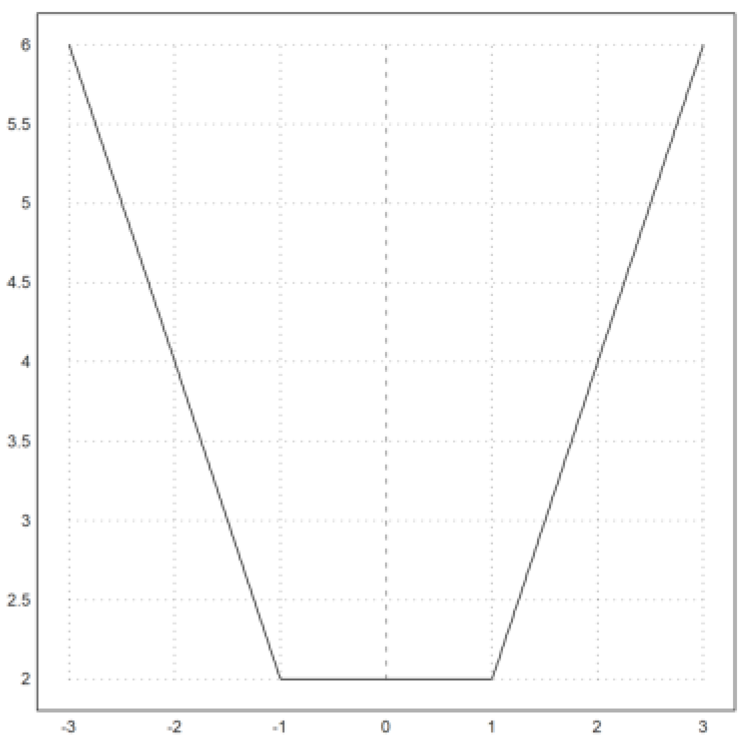
>plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
>reset

0

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



Gambar 495: images/GEOMETRI_Diva%20Nagita_(23030630024)-211.png



Gambar 496: images/GEOMETRI_Diva%20Nagita_(23030630024)-212.png

unicodehyperref hyphensurl [ ]book xcolor amsmath,amssymb iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype [protrusion]basicmath parskip graphicx bookmark xurl same hidelinks, pdfcreator=LaTeX via pandoc

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 latar belakang untuk memahami detailnya.

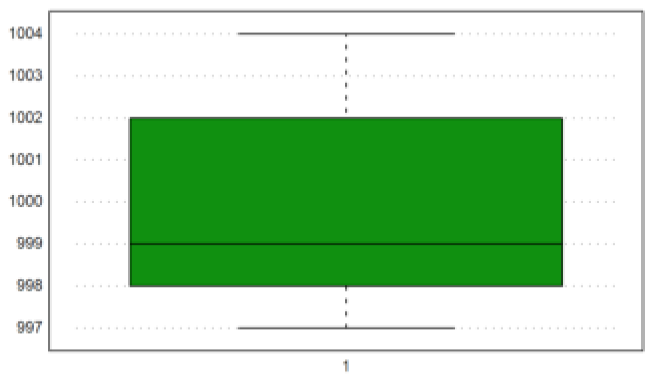
Asumsikan pengukuran berikut ini. Kita ingin menghitung nilai rata-rata dan deviasi standar yang diukur.

>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):



Gambar 497: images/STATISTIKA_Diva%20Nagita_23030630024-001.png

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=" %")

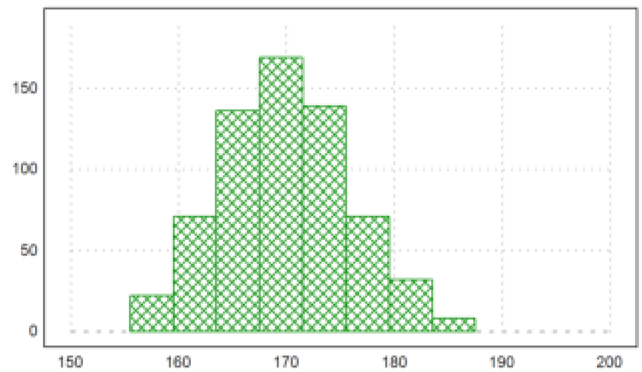
```
3.07 %
```

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="\/"):



Gambar 498: images/STATISTIKA_Diva%20Nagita_23030630024-003.png

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,

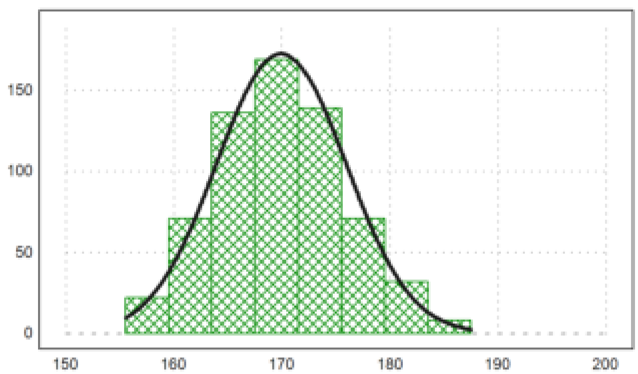
```
169.901234568
5.98912964449
```

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):



Gambar 499: images/STATISTIKA_Diva%20Nagita_23030630024-005.png

Tabel

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 perlu memberikannya 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 statistik lainnya nilai.

>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.

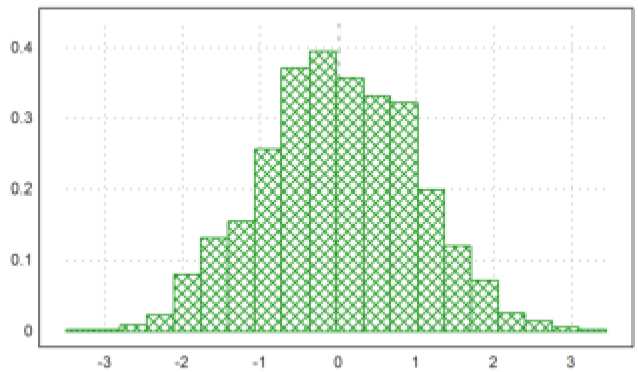
>fileremove(filename);

Distribusi

Dengan plot2d, ada metode yang sangat mudah untuk memplot distribusi data eksperimen.

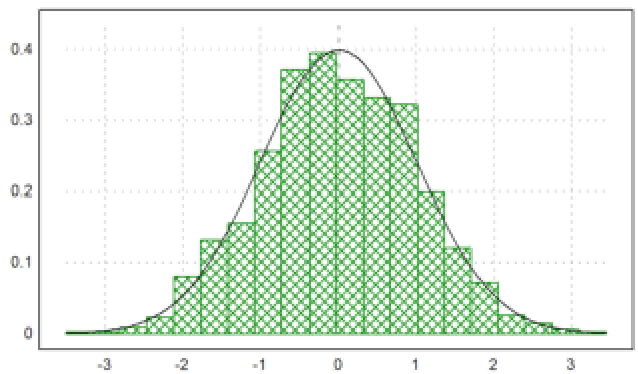
>p=normal(1,1000); //1000 sampel acak berdistribusi normal p

>plot2d(p,distribution=20,style="\/"): // plot sampel acak p



Gambar 500: images/STATISTIKA_Diva%20Nagita_23030630024-006.png

>plot2d("qnormal(x,0,1)",add=1): // tambahkan plot distribusi normal standar



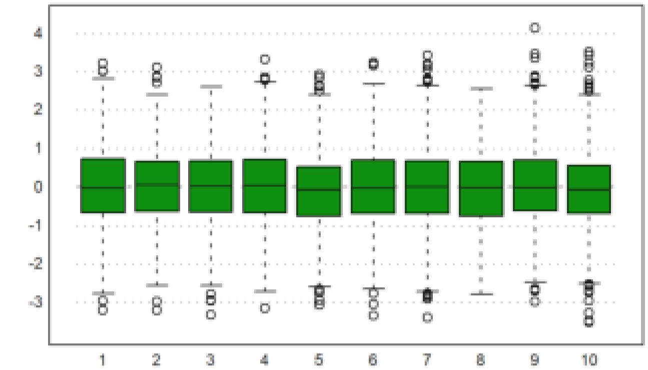
Gambar 501: images/STATISTIKA_Diva%20Nagita_23030630024-007.png

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):

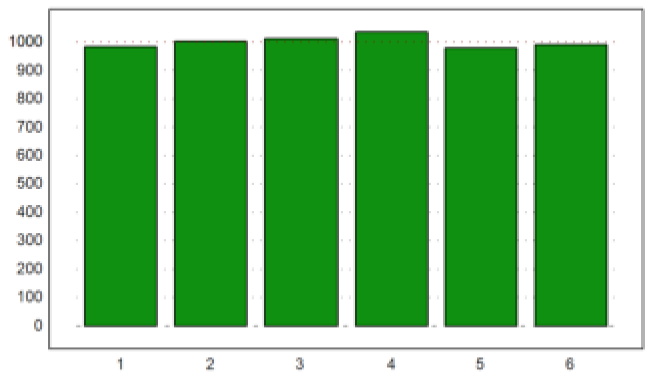
Untuk menghasilkan bilangan bulat acak, Euler memiliki intrandom. Mari kita simulasikan pelemparan dadu dan memplot distribusinya.



Gambar 502: images/STATISTIKA_Diva%20Nagita_23030630024-008.png

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):



Gambar 503: images/STATISTIKA_Diva%20Nagita_23030630024-009.png

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.

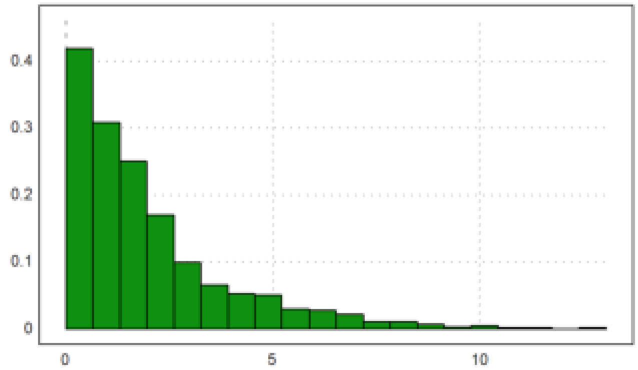
Misalnya, kita mencoba distribusi eksponensial. Sebuah variabel acak kontinu X dikatakan memiliki distribusi eksponensial, jika PDF-nya diberikan oleh

$$f_X(x) = \lambda e^{-\lambda x}, \quad x > 0, \quad \lambda > 0,$$

dengan parameter

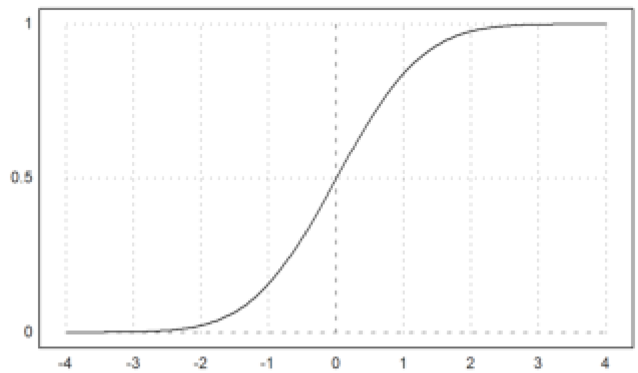
$$\lambda = \frac{1}{\mu}, \quad \mu \text{ adalah mean, dan dilambangkan dengan } X \sim \text{Eksponensial}(\lambda).$$

>plot2d(randexponential(1,1000,2),>distribution):



Gambar 504: images/STATISTIKA_Diva%20Nagita_23030630024-012.png

Untuk banyak distribusi, Euler dapat menghitung fungsi distribusi dan kebalikannya.
>plot2d("normaldis",-4,4):



Gambar 505: images/STATISTIKA_Diva%20Nagita_23030630024-013.png

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
```

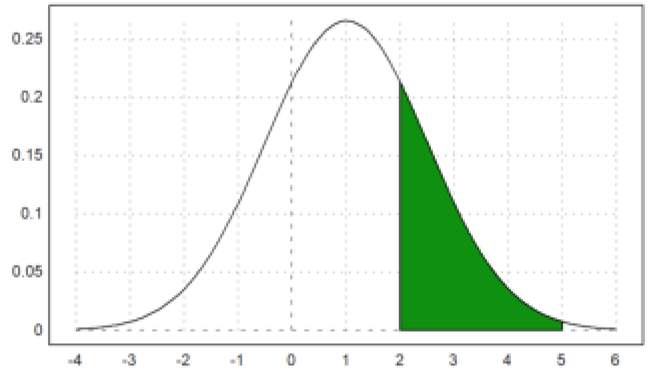
Hal ini dapat dihitung secara numerik dengan integral berikut ini.

$$\int_{2}^{5} \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-1}{1.5})^2}\ dx.$$

>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.



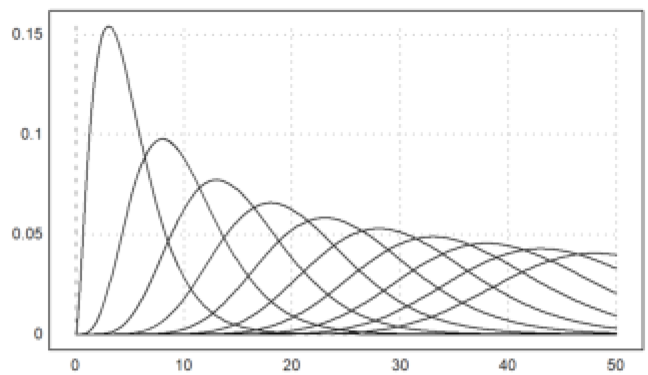
Gambar 506: images/STATISTIKA_Diva%20Nagita_23030630024-014.png

>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):



Gambar 507: images/STATISTIKA_Diva%20Nagita_23030630024-017.png

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
```

Distribusi Diskrit

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]
```

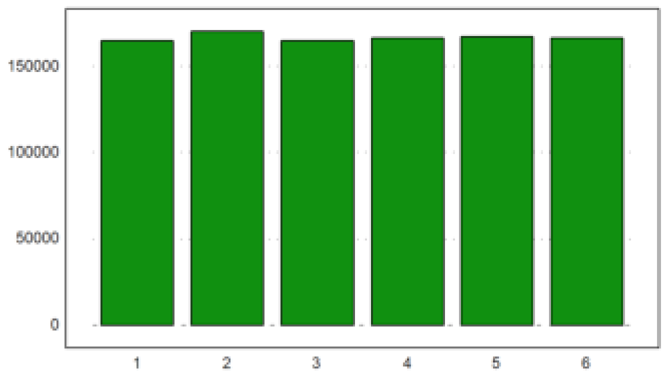
Artinya, dengan probabilitas wd[i+1]-wd[i] kita menghasilkan nilai acak i.

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))):



Gambar 508: images/STATISTIKA_Diva%20Nagita_23030630024-018.png

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.

ution.

>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
```

Memplot Data

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
```

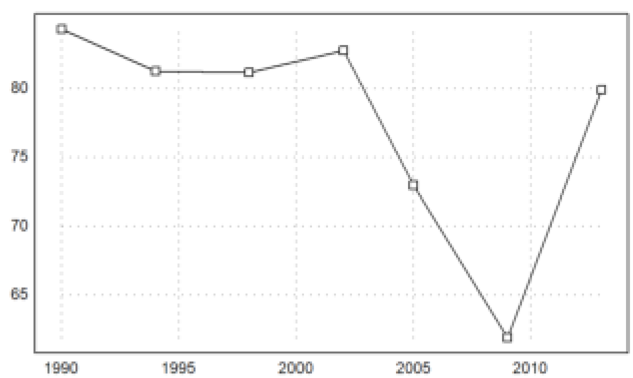
Perkalian matriks berikut ini mengekstrak jumlah persentase dua partai besar yang menunjukkan bahwa partai-partai kecil telah memperoleh suara di parlemen hingga tahun 2009.

>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"):



Gambar 509: images/STATISTIKA_Diva%20Nagita_23030630024-020.png

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.

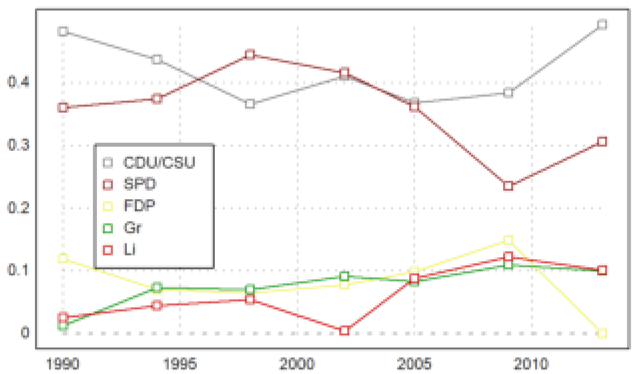
>figure(2,1); . . .
> figure(1); columnsplot(BW[6,3:7],P,color=CP); . . .

> figure(2); columnsplot(BW[6,3:7]-BW[5,3:7],P,color=CP); . . .
> figure(0):

Gambar 510: images/STATISTIKA_Diva%20Nagita_23030630024-021.png

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):



Gambar 511: images/STATISTIKA_Diva%20Nagita_23030630024-022.png

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.

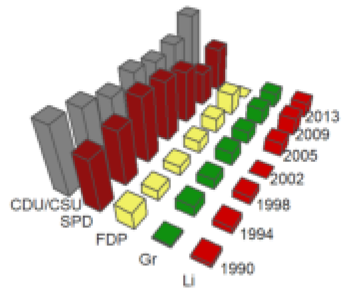
>shrinkwindow(>smaller); . . .
> mosaicplot(BT',srows=YT,scols=P,color=CP,style="#"); . . .
> shrinkwindow():

Kita juga bisa membuat diagram lingkaran. Karena warna hitam dan kuning membentuk sebuah koalisi, kita menyusun ulang elemen-elemennya.

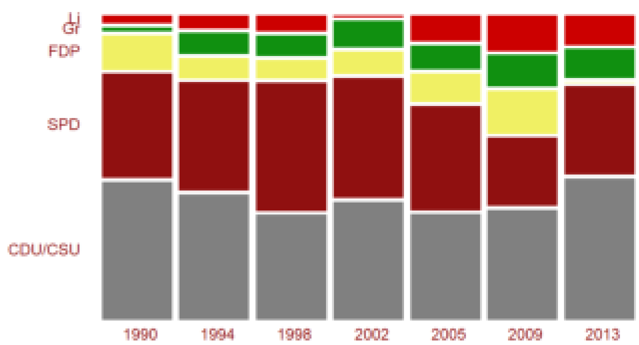
>i=[1,3,5,4,2]; piechart(BW[6,3:7][i],color=CP[i],lab=P[i]):

Berikut ini jenis plot yang lain.

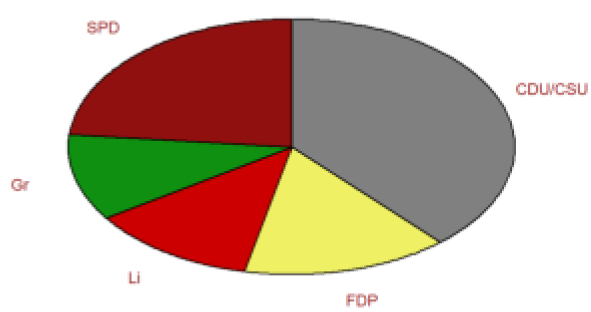
>starplot(normal(1,10)+4,lab=1:10,>rays):



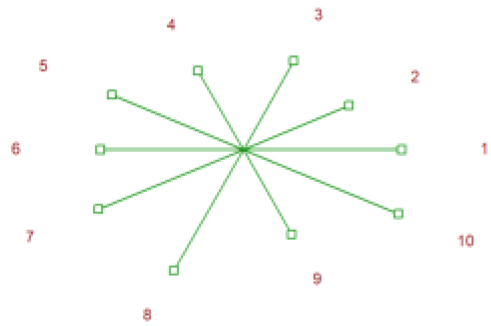
Gambar 512: images/STATISTIKA_Diva%20Nagita_23030630024-023.png



Gambar 513: images/STATISTIKA_Diva%20Nagita_23030630024-024.png



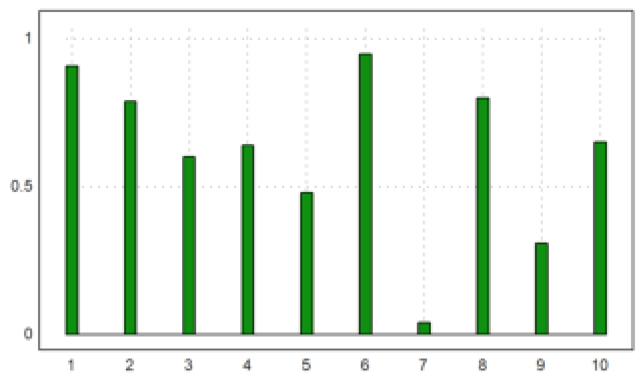
Gambar 514: images/STATISTIKA_Diva%20Nagita_23030630024-025.png



Gambar 515: images/STATISTIKA_Diva%20Nagita_23030630024-026.png

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):



Gambar 516: images/STATISTIKA_Diva%20Nagita_23030630024-027.png

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.

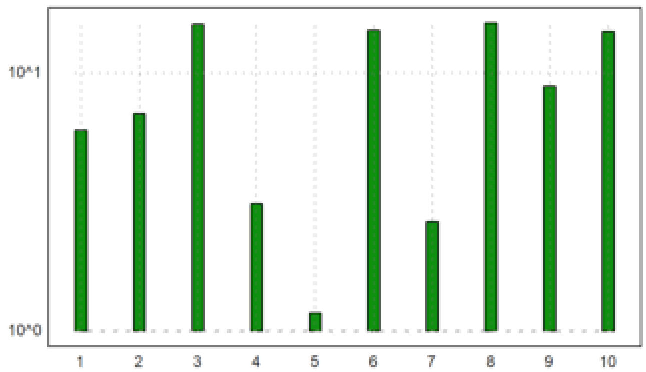
>v=strtochar("the quick brown fox jumps over the lazy dog"); . . .
> w=ascii("a"):ascii("z"); x=getmultiplicities(w,v); . . .
> cw=[]; for k=w; cw=cw | char(k); end; . . .
> columnsplot(x,lab=cw,width=0.05):

Anda juga dapat menetapkan sumbu secara manual.

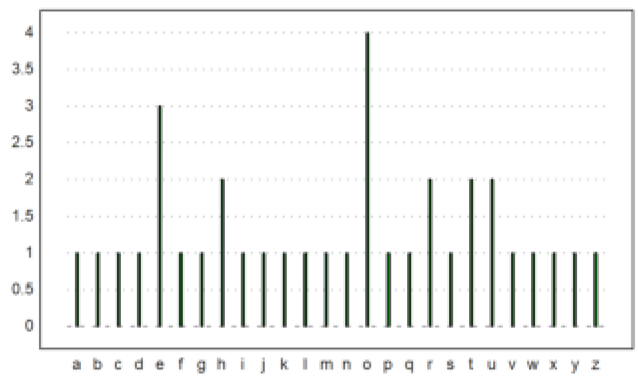
>n=10; p=0.4; i=0:n; x=bin(n,i)*$p^{i*(1-p)}$(n-i); . . .
> columnsplot(x,lab=i,width=0.05,<frame,<grid); . . .
> yaxis(0,0:0.1:1,style="->",>left); xaxis(0,style="."); . . .
> label("p",0,0.25), label("i",11,0); . . .
> textbox(["Binomial distribution","with p=0.4"]):

Berikut ini adalah cara untuk memplot frekuensi angka dalam vektor.

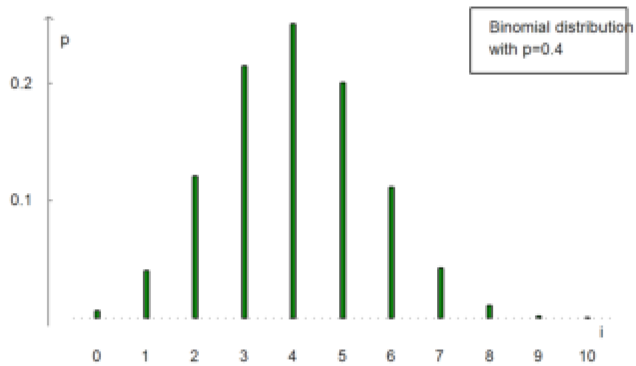
Kami membuat vektor angka acak bilangan bulat 1 hingga 6.



Gambar 517: images/STATISTIKA_Diva%20Nagita_23030630024-028.png



Gambar 518: images/STATISTIKA_Diva%20Nagita_23030630024-029.png



Gambar 519: images/STATISTIKA_Diva%20Nagita_23030630024-030.png

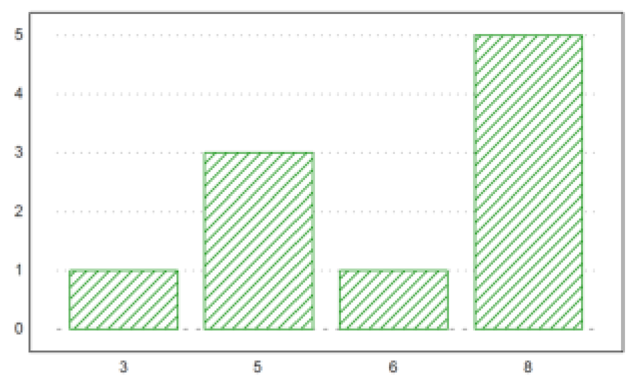
>v:=intrandom(1,10,10)

```
[8,   5,   8,   8,   6,   8,   8,   3,   5,   5]
```

Kemudian ekstrak nomor unik dalam v.
>vu:=unique(v)

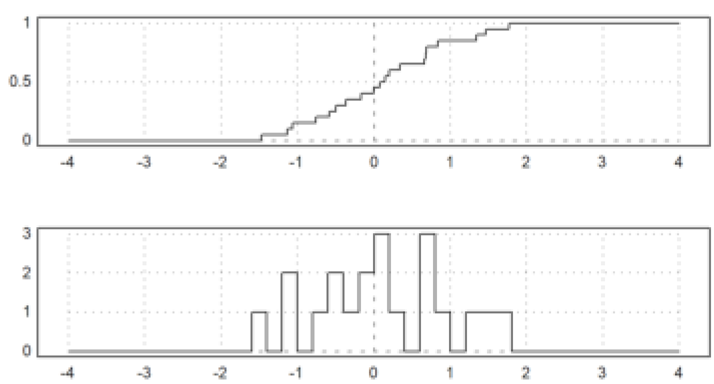
```
[3,   5,   6,   8]
```

Dan memplot frekuensi dalam plot kolom.
>columnsplot(getmultiplicities(vu,v),lab=vu,style="/"):



Gambar 520: images/STATISTIKA_Diva%20Nagita_23030630024-031.png

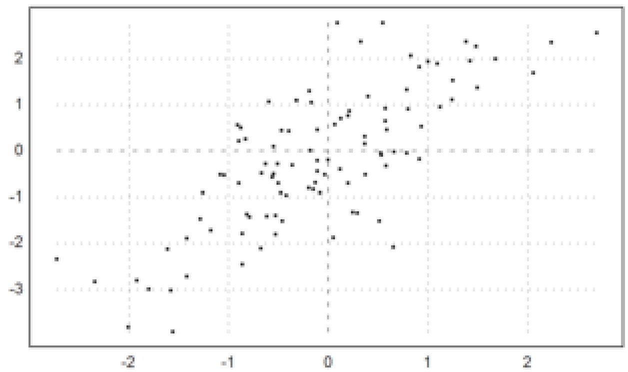
Kami ingin mendemonstrasikan fungsi untuk distribusi nilai empiris.
>x=normal(1,20);
Fungsi empdist(x,vs) membutuhkan larik nilai yang telah diurutkan. Jadi kita harus mengurutkan x sebelum dapat menggunakannya.
>xs=sort(x);
Kemudian kita memplot distribusi empiris dan beberapa batang kepadatan ke dalam 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):



Gambar 521: images/STATISTIKA_Diva%20Nagita_23030630024-032.png

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="..");

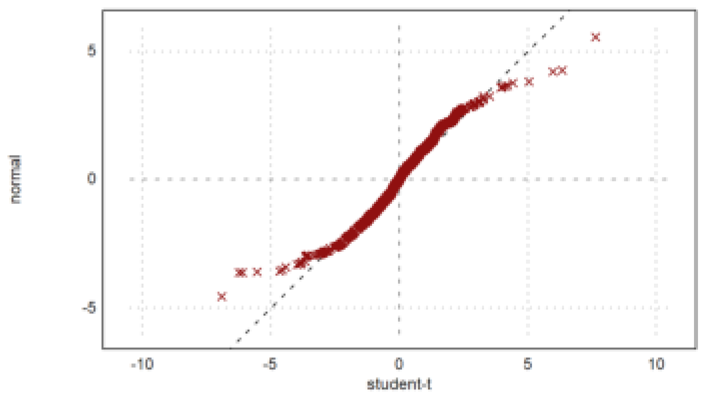


Gambar 522: images/STATISTIKA_Diva%20Nagita_23030630024-033.png

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):



Gambar 523: images/STATISTIKA_Diva%20Nagita_23030630024-034.png

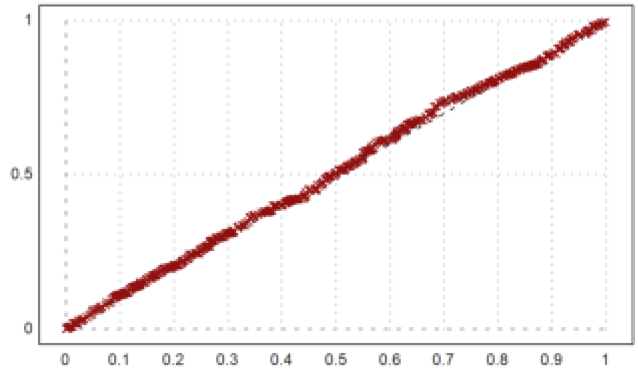
Plot ini dengan jelas menunjukkan bahwa nilai yang terdistribusi normal cenderung lebih kecil pada ujung yang ekstrim.

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):



Gambar 524: images/STATISTIKA_Diva%20Nagita_23030630024-035.png

Regresi dan Korelasi

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.

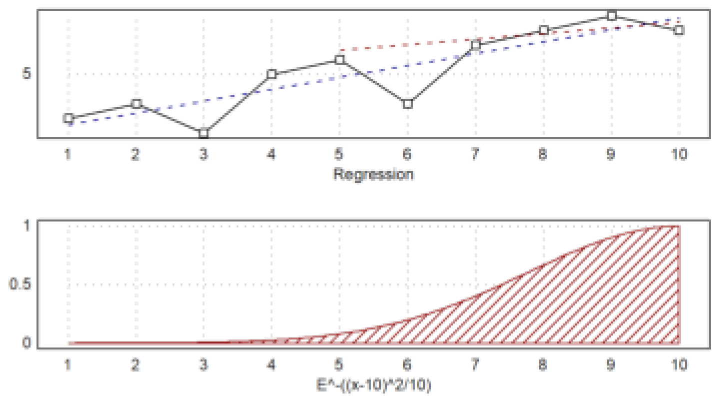
>w &= "exp(-(x-10)^2/10)"; pw=polyfit(x,y,1,w=w(x))

```
[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.



Gambar 525: images/STATISTIKA_Diva%20Nagita_23030630024-036.png

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.

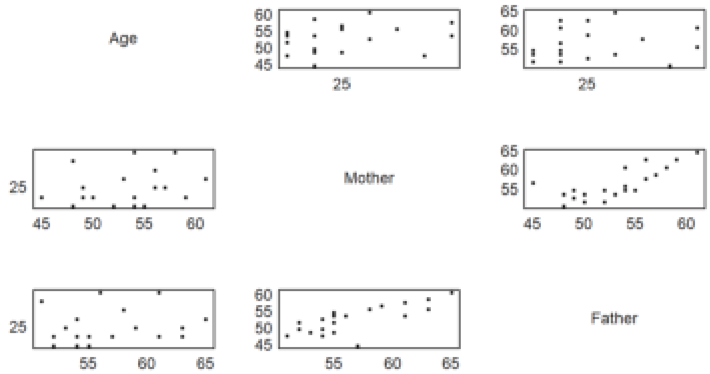
>cs:=MS[,4:5]’; ps:=polyfit(cs[1],cs[2],1)

```
[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, kami menduga fungsi seperti s = a + t. Kemudian a adalah rata-rata dari s-t. Ini adalah perbedaan usia rata-rata antara ayah dan ibu.

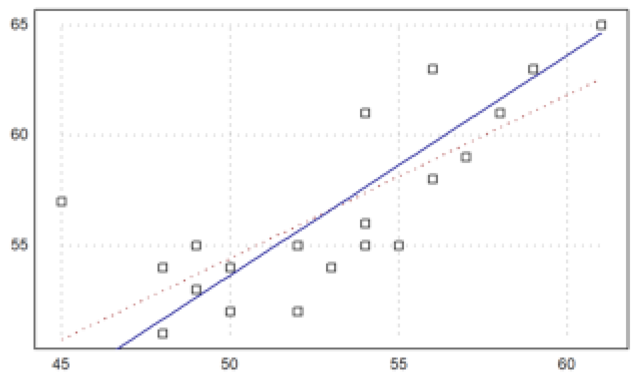
>da:=mean(cs[2]-cs[1])



Gambar 526: images/STATISTIKA_Diva%20Nagita_23030630024-037.png

```
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):



Gambar 527: images/STATISTIKA_Diva%20Nagita_23030630024-038.png

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])

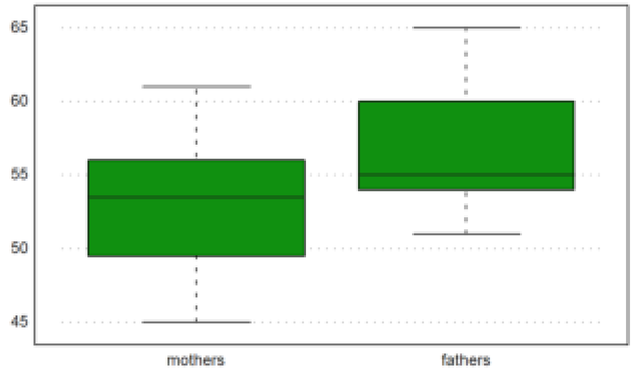
```
1.5
```

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
```



Gambar 528: images/STATISTIKA_Diva%20Nagita_23030630024-039.png

Membuat Fungsi baru

Tentu saja, bahasa EMT dapat digunakan untuk memprogram fungsi baru. Misalnya, kita mendefinisikan fungsi kemiringan.

$$\text{sk}(x) = \frac{\sqrt{n}\sum_i (x_i - m)^3}{(\sum_i (x_i - m)^2)^{3/2}}$$

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
```

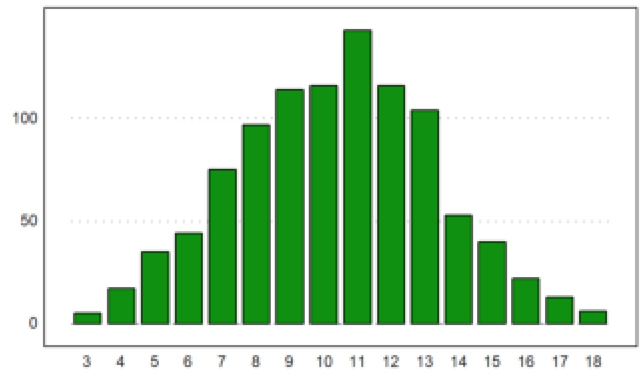
Simulasi Monte Carlo

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.



Gambar 529: images/STATISTIKA_Diva%20Nagita_23030630024-041.png

>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/6ˆ3*1000,>add); plot2d(cw/6ˆ3*1000,>points,>add):

Untuk simulasi lainnya, deviasi nilai rata-rata dari n variabel acak berdistribusi normal 0-1 adalah

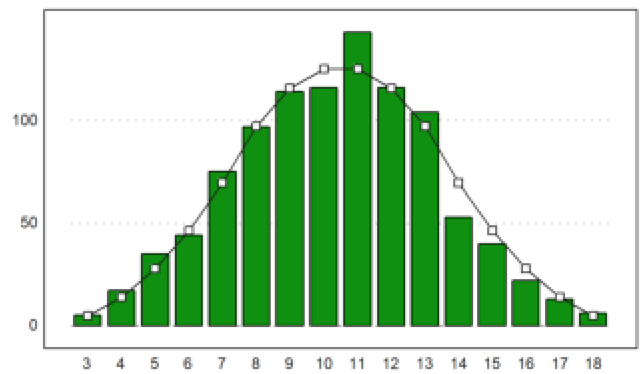
$$\frac{1}{\sqrt{n}}.$$

>longformat; 1/sqrt(10)

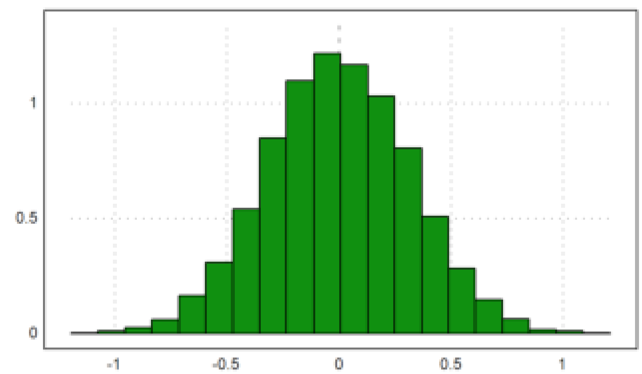
```
0.316227766017
```

Mari kita periksa hal ini dengan sebuah simulasi. Kami menghasilkan 10.000 kali 10 vektor acak.

>M=normal(10000,10); dev(mean(M)')



Gambar 530: images/STATISTIKA_Diva%20Nagita_23030630024-042.png



Gambar 531: images/STATISTIKA_Diva%20Nagita_23030630024-044.png

```
0.319493614817
```

>plot2d(mean(M)',>distribution):

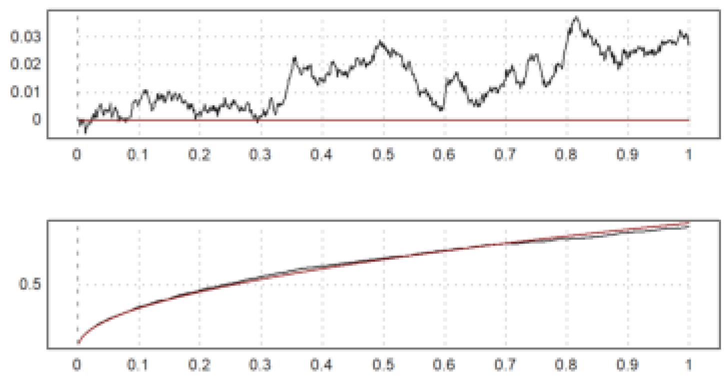
Median dari 10 bilangan acak berdistribusi normal 0-1 memiliki deviasi yang lebih besar.

>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):



Gambar 532: images/STATISTIKA_Diva%20Nagita_23030630024-045.png

Tes

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
```

Fungsi ttest() membutuhkan nilai rata-rata, deviasi, jumlah data, dan nilai rata-rata untuk diuji.

Sekarang mari kita periksa dua pengukuran untuk mean yang sama. Kita tolak hipotesis bahwa kedua pengukuran tersebut memiliki nilai rata-rata yang sama, jika hasilnya < 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
```

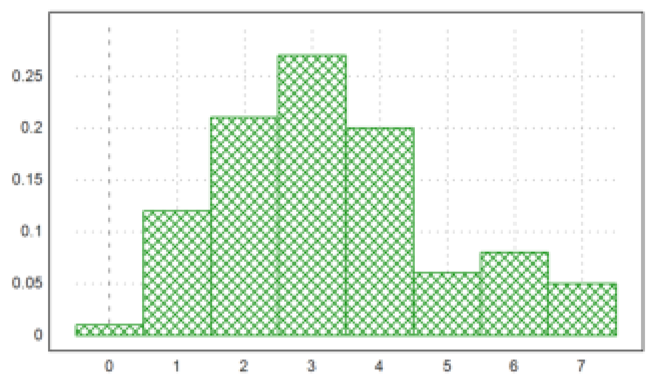
Pada contoh berikut, kita membuat 20 lemparan dadu secara acak sebanyak 100 kali dan menghitung jumlah dadu yang muncul. Rata-rata harus ada 20/6 = 3,3 mata dadu.

>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="\/"):



Gambar 533: images/STATISTIKA_Diva%20Nagita_23030630024-046.png

>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 yang telah dikoreksi. Karena koefisien ini sangat dekat dengan 0, kami menyimpulkan bahwa pemungutan suara tidak bergantung pada jenis kelamin.

>contingency(A)

```
0.0427225484717
```

Beberapa Tes Lainnya

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
```

Bilangan Acak

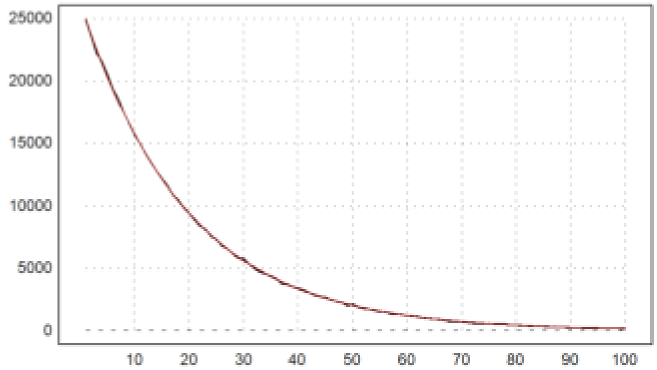
Berikut ini adalah tes untuk generator bilangan acak. Euler menggunakan generator yang sangat bagus, jadi kita tidak perlu mengharapkan adanya masalah.

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));



Gambar 534: images/STATISTIKA_Diva%20Nagita_23030630024-047.png

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"© 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
```

Pengindeksan

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.

```
[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)]; ... ...
                ^
[10.4,  3.1]
```

Tipe Data

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.

>v=zeros(10000); for i=1 to 10000; v[i]=i; end;

To change date types in EMT, you can use functions like 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
```

Larik

EMT hanya memiliki dua dimensi untuk array. Tipe datanya disebut matriks. Akan lebih mudah untuk menulis fungsi untuk dimensi yang lebih tinggi atau sebuah 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]
```

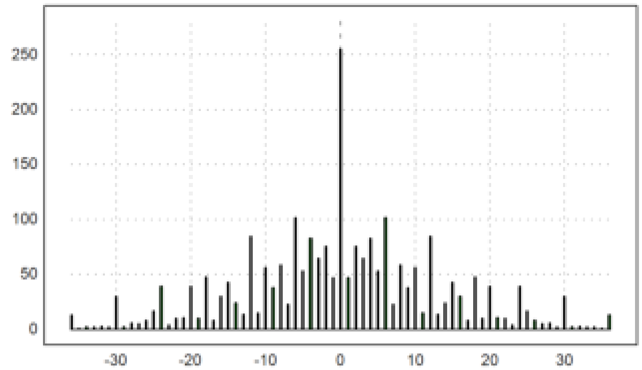
Cara yang paling mudah untuk memplot ini sebagai distribusi adalah sebagai berikut.

>plot2d(q,distribution=11):

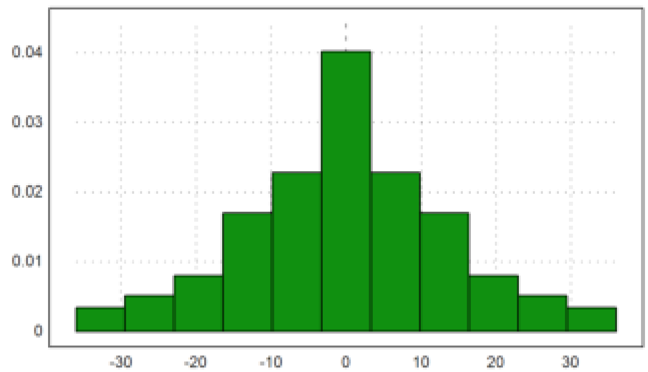
Tetapi juga memungkinkan untuk menghitung jumlah dalam interval yang dipilih sebelumnya. Tentu saja, berikut ini menggunakan getfrequencies() secara internal.

Karena fungsi histo() mengembalikan frekuensi, kita perlu menskalakannya sehingga integral di bawah grafik batang adalah 1.

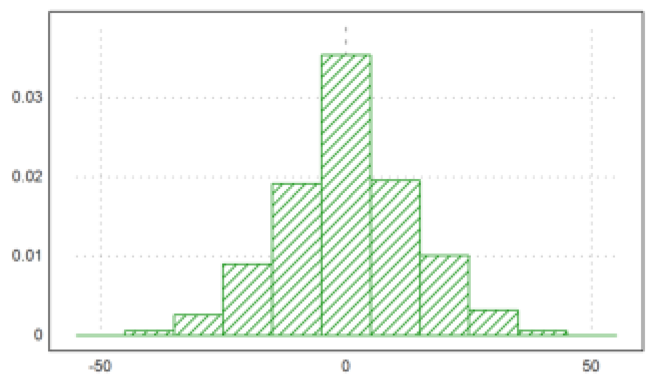
>{x,y}=histo(q,v=-55:10:55); y=y/sum(y)/differences(x); ...
> plot2d(x,y,>bar,style="/"):



Gambar 535: images/STATISTIKA_Diva%20Nagita_23030630024-048.png



Gambar 536: images/STATISTIKA_Diva%20Nagita_23030630024-049.png



Gambar 537: images/STATISTIKA_Diva%20Nagita_23030630024-050.png

Daftar

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, kami 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.

Misalnya, 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
```

The function readtable() in its simplest form can read this and return a collection of values and heading lines.

>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
```

File CSV

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();

The matrices are separated by a blank line. To read the matrices, open the file and call readmatrix() several times.

>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 "simpan sebagai" 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.

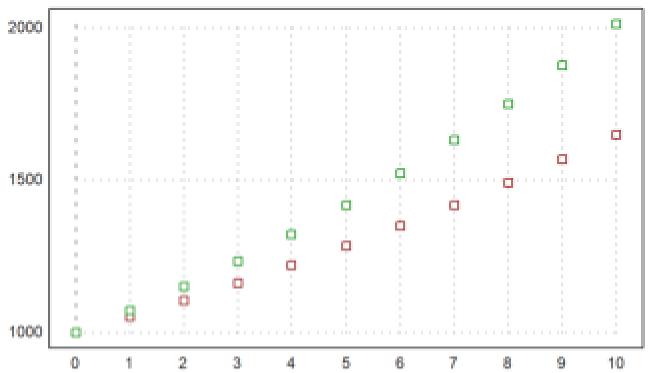
Cara termudah untuk membaca ini ke dalam Euler adalah readmatrix(). Semua koma digantikan oleh titik dengan parameter >comma. Untuk CSV bahasa Inggris, hilangkan saja parameter ini.

>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]'):



Gambar 538: images/STATISTIKA_Diva%20Nagita_23030630024-051.png

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 di 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
```

Menggunakan Tabel

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
```

Menganalisis Garis

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
```

Pertama, kita dapat memberi tanda pada garis tersebut.

>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>"

```
&lt;tr&gt;&lt;td&gt;1145.45&lt;/td&gt;&lt;td&gt;5.6&lt;/td&gt;&lt;td&gt;-4.5&lt;/td&gt;&lt
```

Untuk mengekstrak ini, kita menggunakan ekspresi reguler, yang mencari

- tanda kurung tutup >,

- string apa pun 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,">([ˆ<\>]+)<.+?>([ˆ<\>]+)<");

Hasilnya adalah posisi kecocokan, string yang cocok, dan vektor string untuk sub-cocokan.

>for k=1:length(vt); vtk, 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
```

Membaca dari Web

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
```

Masukan dan Keluaran Variabel

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, 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. National Advisory Committte bagian imunisasi di Canada menyediakan rekomendasi kesehatan tertentu dan laporan ringkasan. Suatu sampel acak pasien-pasien yang dites positif influenza selama musim flu tahun 2012-2013 dicatat. Masing-masing orang diklasifikan dalam grup umur dan jenis flu. Ringkasan data disajikan dalam tabel berikut:

>B=[207,849,1592,617;66,630,754,895;319,1207,1617,512;300,1196,1946,510;120,3599,5860,597]; . . .
> writetable(B,wc=6,labr=["KDL","LEP","DPEE","ELEE","ELP"],labc=1:4)

```
           1     2     3     4
   KDL   207   849  1592   617
```

```
   LEP    66   630   754   895
  DPEE   319  1207  1617   512
  ELEE   300  1196  1946   510
   ELP   120  3599  5860   597
```

>contingency(B)

```
0.35916
```

>writetable(expectedtable(B),wc=6,dc=1,labr=["KDL","LEP","DPEE","ELEE","ELP"],labc=1:4)

```
           1     2     3     4
   KDL 141.21044.11642.6   437
   LEP 101.4 749.91179.8 313.9
  DPEE 158.11168.91838.8 489.2
  ELEE   1711263.81988.2 528.9
   ELP 440.23254.25119.5  1362
```

>contingency(expectedtable(B))

```
0
```

Oleh karena (p-value < 0.01) maka H_0 ditolak, sehingga pada taraf signifikansi 0.01 dapat disimpulkan bahwa ada bukti bahwa grup umur dan jenis influenza adalah dependen.

2. Suatu studi penelitian menunjukkan bahwa diet tinggi garam pada wanita tua meningkatkan risiko patah tulang. Meskipun mekanisme biologisnya masih tidak jelas, namun terlihat ada hubungan antara asupan natrium yang berlebihan dan kerapuhan tulang. Salah satu ukuran kesehatan tulang adalah kadar vitamin D dalam darah. Seperti yang ditentukan oleh kuesioner makanan, sampel acak independen dari wanita tua dalam empat kategori asupan garam diperoleh. Kadar vitamin D dalam darah (dalam nmol / L) diukur di masing-masing. Data diberikan dalam tabel berikut. Adakah bukti yang menunjukkan bahwa setidaknya dua dari rata-rata populasi kadar vitamin D dalam darah berbeda? Gunakan alpha = 0.05.

>a1=[91.5,77.5,94.5,77.5,92.0]; mean(a1),

```
86.6
```

>a2=[89.0,92.0,98.2,80.0,86.7]; mean(a2),

```
89.18
```

>a3=[92.5,100.7,94.0,93.3,106.3]; mean(a3),

```
97.36
```

>a4=[100.1,98.0,99.1,103.9,97.6]; mean(a4),

```
99.74
```

>varanalysis(a1,a2,a3,a4)

```
0.011657
```

Oleh karena F = 5.08 > 3.24 dan p-value <0.05 maka H_0 ditolak. Jadi pada taraf signifikansi 0.05 dapat disimpulkan ada bukti bahwa minimal ada dua rata-rata populasi kadar vitamin D dalam darah yang berbeda.

3. Bagi peternak lebah yang ingin memanen lebih banyak madu, ada empat kemungkinan untuk mendapatkan lebih banyak lebah: package bees, nucs, colonies, swarms. Departemen ilmu pertanian dari universitas tertentu memperoleh sampel acak dari pembelian lebah, dan masing-masing diklasifikasikan ke dalam salah satu dari empat kategori tersebut. Gunakan tabel frekuensi satu arah berikut untuk menguji hipotesis bahwa empat kemungkinan pembelian lebah terjadi dengan frekuensi yang sama. Gunakan taraf signifikansi Alpha = 0.05

>chitest([31,36,26,20],dup(28.25,4)')

```
0.1731
```

Diperoleh

$$\text{p-value} = P(X^2 > 4.9823) = 0.1731]Olehkarena$$

$X^2 = 4.9823 < 7.8147 ataup{-}value = 0.1731 > 0.05]makaH\_0tidakditolak.Padatarafsignifikansialpha = 0.05, tidakadab$

4. Dengan taraf nyata 5% ujilah apakah (peringkat) pendapatan di departemen Q lebih kecildibandingkan departemen Z?

>p1=[6,10,15,32];
>p2=[12,13,15,15,20,31,38,40];
>ranktest(p1,p2)

```
0.11724
```

Karena didapat z hitung = -1.19 ada di daerah penerimaan H_0 maka H_0 diterima, sehingga peringkat Pendapatan di kedua departemen sama.

5. Untuk melihat apakah ada perbedaan produksi per hektar tanaman jagung karena pengaruh duametode penanaman yang digunakan, pertumbuhan tanaman jagung dipilih dari sejumlah plottanah yang berbeda secara random. Kemudian produksi per hektar dari masing-masing plot dihitung dan hasilnya adalah sebagai berikut: (? = 5%)

Metode 1 : 83 91 94 89 96 91 92 90 92 85
Metode 2 : 91 90 81 83 84 83 88 91 90 84 80 85
Apakah dua metode tersebut memiliki nilai median yang sama?
>b1=[83,91,94,89,96,91,92,90,92,85];
>b2=[91,90,81,83,84,83,88,91,90,84,80,85];
>mediantest(b1,b2)

```
0.98501
```

Harga Chi kuadrat tabel dk =1 dan alpha 5%= 3.841 karena Chi hitung < chi kuadrat tabel maka H_0 tidak ditolak. Sehingga disimpulkan dua metode mempunyai nilai median yang sama untuk produksi per hektar.