

TUGAS PEKAN 16 APLIKASI KOMPUTER

BAB LaTeX dan Markdown



Shalih Abdillah
22305144009
Matematika B 2022

**PRODI MATEMATIKA
DEPARTEMEN PENDIDIKAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI YOGYAKARTA
2023**

DAFTAR ISI

1 KB Pekan 2 : Belajar Menggunakan Software EMT	2
2 KB Pekan 3 : Menggunakan EMT untuk menyelesaikan masalah-masalah Aljabar	17
3 KB Pekan 4: Menggunakan EMT untuk menggambar grafik 2 dimensi (2D)	81
4 KB Pekan 5: Menggunakan EMT untuk menggambar grafik 3 dimensi (3D)	164
5 KB Pekan 6-7: Menggunakan EMT untuk kalkulus	421
6 KB Pekan 8: Menggunakan EMT untuk Geometri	495
7 KB Pekan 10; Menggunakan EMT untuk Statistika	559

BAB 1

KB PEKAN 2 : BELAJAR MENGGUNAKAN SOFTWARE EMT

article

eumat

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 open source 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 Tutorials and Examples", 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 meng-klik 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) = \int_a^x f (t) \, dt
- mathjax: \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

$$F(x) = \int_a^x f(t) dt$$
$$\frac{x^2 - 1}{x - 1} = x + 1$$

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

Gambar diambil dari folder images di tempat file notebook berada dan tidak dapat dibaca dari Web. Untuk "See:", tautan (URL)web lokal dapat digunakan.

Paragraf terdiri atas satu baris panjang di editor. Pergantian baris akan memulai baris baru. Paragraf harus dipisahkan dengan baris kosong.

```
>// 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 untuk Anda

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

Jawab

```
> p=3; l=4; p*l, 2*(p+l)
```

```
12  
14
```

```
>k=12; s=3; k/s, k+s, k-s, s-k, s/k
```

```
4  
15  
9  
-9  
0.25
```

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 \circ . 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 untuk Anda

- 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.

Jawab

```
>a= [1,3,5,6,7] // vektor atau matriks baris
```

[1, 3, 5, 6, 7]

```
>c = [3,2,4,2; 1,2,3,4; 8,7,9,4; 6,1,3,7] // penulisan matriks
```

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

```
>(1+sqrt(25))/3 //sqrt=akar kuadrat
```

2

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 mengatramilan 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 "deformat" atau "reset".

```
>longestformat; pi, deformat; 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*x^2+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 Variabel

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

```
>listvar
```

r	1.25
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

$$i = \sqrt{-1}$$

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 kompleks 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°
```

```
>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
```

$$(b + a)^2$$

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

$$b^2 + 2ab + a^2$$

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

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

$$[x = \frac{(-\sqrt{b^2 - 4ac}) - b}{2a}, x = \frac{\sqrt{b^2 - 4ac} - b}{2a}]$$

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

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

$$a^2 - b^2$$

```
>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)
```

$$\frac{3}{2}^3 \cdot 5^3$$

```
>::: 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)!)
```

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

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

$$\cos(x) \cos(y) - \sin(x) \sin(y)$$

```
>::: trigexpand(sin(x+y));
```

$$\cos(x) \sin(y) + \sin(x) \cos(y)$$

```
>::: trigsimp(((1-sin(x)^2)*cos(x))/cos(x)^2+tan(x)*sec(x)^2) //menyederhanakan fungsi tri
```

$$\frac{\sin^4(x) + \cos^4(x)}{\cos^3(x)}$$

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

```
>p1 &= (x^3+1) / (x+1)
```

$$\begin{array}{r} 3 \\ x + 1 \\ \hline x + 1 \end{array}$$

```
>&ratsimp(p1)
```

$$x^2 - x + 1$$

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
```

$$\begin{array}{r} 3 \\ (b + a)^3 + 1 \\ \hline b + a + 1 \end{array}$$

$$b^2 + (2ab - 1)b + a^2 - a + 1$$

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

$$\begin{array}{r} 2 \quad 3 \\ 3x^2 - x^3 + 1 \\ \hline x^2 - (x^2 + 1) \end{array}$$

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

$$\begin{array}{r} 3 \quad \quad \quad 2 \\ 2x^3 - 3x^2 + 6x \\ \hline 6 \end{array}$$

Tampilan Matematika Simbolik dengan LaTeX

Anda dapat menampilkan hasil perhitungan 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)
```

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

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

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

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

$$a - b$$

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.

```
> $&integrate((x-11)/(x^2+3*x-10),x)
```

$$\frac{16 \log(x + 5)}{7} - \frac{9 \log(x - 2)}{7}$$

```
> $&integrate((cos(t))/((sin(t))^4-16),t)
```

$$-\frac{\log(\sin t + 2)}{32} - \frac{\arctan\left(\frac{\sin t}{2}\right)}{16} + \frac{\log(\sin t - 2)}{32}$$

```
> $&diff(x^2*sin(x*y^2),x)
```

$$2x \sin(xy^2) + x^2 y^2 \cos(xy^2)$$

```
> $&diff(x^2*sin(x*y^2),y)
```

$$2x^3 y \cos(xy^2)$$

BAB 2

KB PEKAN 3 : MENGGUNAKAN EMT UNTUK MENYELESAIKAN MASALAH-MASALAH ALJABAR

article
eumat

EMT untuk Perhitungan Aljabar

Shalih Abdillah
22305144009
Matematika B 2022

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

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

Contoh pertama

Menyederhanakan bentuk aljabar:

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

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

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

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

$$-\frac{9}{y^2}$$

Menjabarkan:

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

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

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

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

$$\text{showev} \cdot \text{expand}\left(\frac{-2x^5 y^4 - (x^3 + 8x^2)y^2 - 4}{x^3 y^2}\right)$$

Baris Perintah

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

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

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

16.7551608191

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

113.097335529

Perintah harus dipisahkan dengan yang kosong. Baris perintah berikut mencetak dua hasilnya.

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

```
37.6991118431  
75.3982236862
```

```
>pi*r*h; %+pi*r*h
```

```
37.6991118431
```

Baris perintah dieksekusi dalam urutan yang ditekan pengguna kembali. Jadi Anda mendapatkan nilai baru setiap kali Anda menjalankan baris kedua.

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

```
0.540302305868
```

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

```
0.857553215846
```

```
>y := 3;  
>y := cos(y)
```

```
-0.9899924966
```

Jika dua garis terhubung dengan "..." kedua garis 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
```

```
>y := 5; ...  
>y := (y+3/y), y := (y+3/y)/3, y := (y+3/3)
```

```
5.6  
2.04523809524  
3.04523809524
```

Ini juga merupakan cara yang baik untuk menyebarkan perintah panjang pada dua atau lebih baris. Anda dapat menekan Ctrl+Return untuk membagi garis menjadi dua pada posisi kursor saat ini, atau Ctrl+Back untuk menggabungkan garis.

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

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

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

81

```
>%+ y=7^2; ...
```

54

Euler mendukung loop di baris perintah, selama mereka masuk ke dalam satu baris atau multi-baris. Dalam program, pembatasan ini tidak berlaku, tentu saja. Untuk informasi lebih lanjut lihat pengantar berikut.

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

```
1.5  
1.4166666667  
1.41421568627  
1.41421356237  
1.41421356237
```

```
>x=2; for i=1 to 6; x := (x+2/x-1)/2, end; // menghitung akar 2
```

```
1  
1  
1  
1  
1  
1
```

Tidak apa-apa untuk menggunakan multi-line. 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 berfungsi.

```
>if E^pi>pi^E; then "Kupikir Begitu!", endif;
```

```
Kupikir Begitu!
```

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

Saat Anda menggerakkan kursor di sepanjang garis, pasangan tanda kurung atau kurung buka dan tutup akan disorot. Juga, perhatikan baris status. Setelah kurung buka fungsi `sqrt()`, baris status akan menampilkan teks bantuan untuk fungsi tersebut. Jalankan perintah dengan tombol kembali.

```
>sqrt(sin(10°)/cos(20°))
```

0.429875017772

```
>sqrt(sin(20°)/cos(30°))
```

0.62843523424

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

Anda dapat mengklik dua kali pada perintah apa pun untuk membuka bantuan untuk perintah ini. Coba klik dua kali perintah `exp` di bawah ini di baris perintah.

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

2.5

```
>exp(log(2.7))
```

2.7

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

Euler tahu fungsi matematika biasa. Seperti yang Anda lihat di atas, fungsi trigonometri bekerja dalam radian atau derajat. Untuk mengonversi ke derajat, tambahkan simbol derajat (dengan tombol F7) ke nilainya, atau gunakan fungsi `rad(x)`. Fungsi akar kuadrat disebut kuadrat dalam Euler. Tentu saja, $x^{(1/2)}$ juga dimungkinkan.

Untuk menyetel variabel, gunakan `=` atau `:=`. Demi kejelasan, pengantar ini menggunakan bentuk yang terakhir. Spasi tidak masalah. Tapi ruang antara perintah diharapkan.

Beberapa perintah dalam satu baris dipisahkan dengan `"` atau `";`. Titik koma menekan output dari perintah. Di akhir baris perintah `"` diasumsikan, jika `";` hilang.

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

30.65625

EMT menggunakan sintaks pemrograman untuk ekspresi. Memasuki

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

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

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

8.77908249441

Untuk menghitung ekspresi rumit seperti

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

Anda harus memasukkannya dalam bentuk baris.

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

23.2671801626

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

Hasil dari perhitungan ini adalah bilangan floating point. Secara default dicetak dengan akurasi sekitar 12 digit. Di baris perintah berikut, kita juga belajar bagaimana kita bisa 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, itu harus mengandung tanda kurung untuk memaksa urutan eksekusi yang benar. Jika ragu, memasang braket adalah ide yang bagus. Perhatikan bahwa EMT menunjukkan tanda kurung buka dan tutup saat mengedit baris perintah.

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

14.4978445072

Operator numerik Euler meliputi

```
+ unary atau operator plus  
- unary atau operator minus  
*, /  
. produk matriks  
a^b daya untuk positif a atau bilangan bulat b (a**b juga berfungsi)  
n! operator faktorial
```

dan masih banyak lagi.

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

```
sin,cos,tan,atan,asin,acos,rad,deg  
log,exp,log10,sqrt,logbase  
bin,logbin,logfac,mod,lantai,ceil,bulat,abs,tanda  
conj,re,im,arg,conj,nyata,kompleks  
beta,betai,gamma,complexgamma,ellrf,ellf,ellrd,elle  
bitand, bitor, bitxor, bitnot
```

Beberapa perintah memiliki alias, mis. Untuk log.

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

```
2  
0.5
```

```
>sin(30°)
```

```
0.5
```

Pastikan untuk menggunakan tanda kurung (kurung bulat), setiap kali 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).

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

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

Bilangan Asli

Tipe data utama dalam Euler adalah bilangan real. 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)
```

```
1.01010101010101010101010101010101010101010101010101010101010101*2^-2
```

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

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

String

Sebuah 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 itu.

```
>"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 cm².

Fungsi `print` juga mengonversi angka menjadi string. Ini dapat mengambil sejumlah digit dan sejumlah tempat (0 untuk keluaran padat), dan secara optimal satu unit.

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

Golden Ratio : 1.61803

```
>"Golden Ratio : " + print((1+sqrt(20))/2,5,0)
```

Golden Ratio : 2.73607

Tidak ada string khusus, yang tidak dicetak. Itu dikembalikan oleh beberapa fungsi, ketika hasilnya tidak masalah. (Ini dikembalikan secara otomatis, jika fungsi tidak memiliki pernyataan pengembalian.)

```
>none
```

Untuk mengonversi string menjadi angka, cukup evaluasi saja. Ini juga berfungsi 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 menghasilkan string seperti itu, gunakan `u"..."` dan salah satu entitas HTML. String Unicode dapat digabungkan seperti string lainnya.

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

```
= 45°
```

```
I
```

Dalam komentar, entitas yang sama seperti `alpha;`, `beta;` dll dapat digunakan. Ini mungkin alternatif cepat untuk Latex. (Lebih detail di komentar di bawah).

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

```
>v=strtochar(u"&Auml; is a German letter")
```

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

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

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

```
Ü is a German letter
```

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

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

```
We have =.
```

Dimungkinkan juga untuk menggunakan entitas numerik.

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

```
Ähnliches
```

Nilai Boolean

Nilai Boolean direpresentasikan dengan `1=true` atau `0=false` 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-kata "dan" dan "atau" hanya dapat digunakan dalam kondisi untuk "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 bukan nol() untuk mengekstrak elemen tertentu dari vektor. Dalam contoh, kami menggunakan isprime bersyarat(n).

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

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

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

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

Format Keluaran

Format output default EMT mencetak 12 digit. Untuk memastikan bahwa kami melihat default, kami mengatur ulang format.

```
>defformat; pi
```

3.14159265359

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

```
>longest pi
```

3.141592653589793

Berikut adalah representasi heksadesimal internal dari bilangan 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). Tapi ini bisa diubah.

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

```
3.1416
```

Fungsi "format terpanjang" mengatur format skalar juga.

```
>longestformat; pi
```

```
3.141592653589793
```

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

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

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

Tetapi format output EMT dapat diatur dengan cara yang fleksibel.

```
>longestformat; pi,
```

```
3.141592653589793
```

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

3.14159

Standarnya adalah deformat().

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

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

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

4.934802200544679

Ada juga operator pendek untuk mencetak hasil dalam format pecahan. Kami sudah menggunakannya di atas.

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

25/12

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

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

-1.110223024625157e-16

Tetapi dengan "format panjang" default Anda tidak akan melihat ini. Untuk kenyamanan, output dari angka yang sangat kecil adalah 0.

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

0

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 itu. Parameter tambahan dapat ditambahkan menggunakan parameter yang ditetapkan.

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

-0.919535764538

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

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

36

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

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

45

Untuk referensi, kami berkomentar bahwa koleksi panggilan (dibahas di tempat lain) dapat berisi ekspresi. Jadi kita bisa membuat contoh di atas sebagai berikut.

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

45

Ekspresi dalam x sering digunakan seperti fungsi.

Perhatikan bahwa mendefinisikan fungsi dengan nama yang sama seperti ekspresi simbolik global menghapus variabel ini untuk menghindari kebingungan antara ekspresi simbolik dan fungsi.

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

12

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

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

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

Bentuk khusus dari 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
```

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 simbolis atau numerik. Jika variabel utama adalah x, ekspresi dapat dievaluasi seperti fungsi.

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

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

```
-0.475
```

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

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

```
-0.425
```

Sebuah ekspresi tidak perlu simbolis. Ini diperlukan, jika ekspresi berisi fungsi, yang hanya diketahui di kernel numerik, bukan di Maxima.

Matematika Simbolik

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

Matematika simbolik terintegrasi dengan mulus ke dalam Euler dengan &. Ekspresi apa pun yang dimulai dengan & adalah ekspresi simbolis. Itu dievaluasi dan dicetak oleh Maxima.

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

```
>$&44!
```

```
26582715747884487680436258110146158903196385280000000000
```

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

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

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

```
2481256778
```

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

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

2481256778

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

Anda akan belajar bahwa yang berikut ini juga berfungsi.

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

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

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

Jika Anda ingin mengganti x dengan nilai tertentu, gunakan "dengan".

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

120

Dengan begitu Anda dapat menggunakan solusi persamaan dalam persamaan lain.

Ekspresi simbolik dicetak oleh Maxima dalam bentuk 2D. Alasan untuk ini adalah bendera simbolis khusus dalam string.

Seperti yang akan Anda lihat pada contoh sebelumnya dan berikut, jika Anda telah menginstal LaTeX, Anda dapat mencetak ekspresi simbolis dengan Lateks. Jika tidak, perintah berikut akan mengeluarkan pesan kesalahan.

Untuk mencetak ekspresi simbolis dengan LaTeX, gunakan \$ di depan & (atau Anda dapat menghilangkan &) sebelum perintah. Jangan menjalankan perintah Maxima dengan \$, jika Anda tidak menginstal 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 menyertakan ekspresi dalam "...". Untuk menggunakan lebih dari ekspresi sederhana adalah mungkin, tetapi sangat tidak disarankan.

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

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

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

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

images/Shalih Abdillah_22305144009_ApKom2-014-lar

Sekali lagi, % mengacu pada hasil sebelumnya.

Untuk mempermudah, kami menyimpan solusi ke variabel simbolik. Variabel simbolik didefinisikan dengan "&=".

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

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

Ekspresi simbolik dapat digunakan dalam ekspresi simbolik lainnya.

```
>${\&factor(diff(fx,x))}
```

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

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!)
```

$$\begin{matrix} 8 & 4 & 2 \\ 2 & 3 & 5 & 7 \end{matrix}$$

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

$$\begin{matrix} 18 & 8 & 4 & 2 \\ 2 & 3 & 5 & 7 & 11 & 13 & 17 & 19 \end{matrix}$$

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

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

$$\begin{matrix} 2 \\ g \end{matrix}$$

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

$$\begin{matrix} 3 & x \\ x & E \end{matrix}$$

$$x^3 e^x$$

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

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

$$\begin{matrix} 5 \\ 125 E \end{matrix}$$

$$125 e^5$$

$$18551.64488782208$$

```
>fx(5)
```

$$18551.6448878$$

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

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

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

$$\begin{matrix} 10 & 5 \\ 1000 E & - 125 E \end{matrix}$$

$$2.20079141499189e+7$$

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

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

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

```
>tex(fx)
```

$$x^3 \sqrt{e^x}$$

Ekspresi simbolik dapat dievaluasi seperti ekspresi numerik.

```
>fx(0.5)
```

$$0.206090158838$$

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

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

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

Penugasan juga bisa bersifat simbolis.

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

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

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

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

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

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

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

$$1.56155281281$$

Nilai numerik dari solusi simbolik dapat dihitung dengan evaluasi hasil simbolis. Euler akan membaca tugas x= dll. Jika Anda tidak memerlukan hasil numerik untuk perhitungan lebih lanjut, Anda juga dapat memberikan Maxima menemukan nilai numerik.

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

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

[-3.23607, 1.23607]

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

Untuk mendapatkan solusi simbolis tertentu, seseorang dapat menggunakan "dengan" dan indeks.

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

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

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

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

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

$$2$$

Ekspresi simbolis dapat memiliki bendera, yang menunjukkan perlakuan khusus di Maxima. Beberapa flag dapat digunakan sebagai perintah juga, yang lain tidak. Bendera ditambahkan dengan "!" (bentuk yang lebih bagus dari "ev(...,flags)")

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

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

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

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

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

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

Fungsi

Dalam EMT, fungsi adalah program yang didefinisikan dengan perintah "fungsi". Ini bisa berupa fungsi satu baris atau fungsi multibaris.

Fungsi satu baris dapat berupa numerik atau simbolis. Fungsi satu baris numerik didefinisikan oleh ":=".

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

Untuk gambaran umum, kami menunjukkan semua kemungkinan definisi untuk fungsi satu baris. Suatu fungsi dapat dievaluasi sama seperti fungsi Euler bawaan lainnya.

```
>f(2)
```

4.472135955

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

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

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

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

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

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

0.786151377757

Secara default, jika Anda perlu menimpa fungsi bawaan, Anda harus menambahkan kata kunci "menimpa". Menimpa fungsi bawaan berbahaya dan dapat menyebabkan masalah untuk fungsi lain tergantung pada fungsi tersebut.

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

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

0.707106781187

Lebih baik kita menghapus redefinisi dosa ini.

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

0.707106781187

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

Menyetelnya akan menimpa nilai default.

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

80

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

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

16

Jika suatu variabel bukan parameter, itu harus global. Fungsi satu baris dapat melihat variabel global.

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

24

Tetapi parameter yang ditetapkan menimpa nilai global.

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

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

20

Fungsi simbolis didefinisikan dengan "&=". Mereka didefinisikan dalam Euler dan Maxima, dan bekerja di kedua dunia. Ekspresi yang mendefinisikan dijalankan melalui Maxima sebelum definisi.

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

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

Fungsi simbolik dapat digunakan dalam ekspresi simbolik.

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

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

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

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

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

178.635099908

Mereka dapat digunakan untuk mendefinisikan fungsi atau ekspresi simbolik lainnya.

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

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

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

0.703467422498

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

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

0.703467422498

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

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

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

$$(x + 2)^n$$

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

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

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

625

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

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

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

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

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

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

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

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

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

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

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

$$x^3 - x$$

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

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

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

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

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

yang tidak dapat dinilai secara simbolis.

Jika kita mendefinisikan kembali fungsi dengan kata kunci "peta" dapat digunakan untuk vektor x. Secara internal, fungsi dipanggil untuk semua nilai x satu kali, dan hasilnya disimpan dalam vektor.

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

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

Fungsi dapat memiliki nilai default untuk parameter.

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

Sekarang fungsi dapat dipanggil dengan atau tanpa parameter "basis".

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

2
6.7

Selain itu, dimungkinkan untuk menggunakan parameter yang ditetapkan.

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

2

Seringkali, kita ingin menggunakan fungsi untuk vektor di satu tempat, dan untuk elemen individual di tempat lain. Ini dimungkinkan dengan parameter vektor.

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

$$y^2 - x y + y + x^2$$

Fungsi simbolik seperti itu dapat digunakan untuk variabel simbolik.

Tetapi fungsinya juga dapat digunakan untuk vektor numerik.

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

17

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

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

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

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

```
0
```

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

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

$$10 (y^2 + x)^3 (9 y^2 + x + 2)$$

Untuk meringkas

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

Memecahkan Ekspresi

Ekspresi dapat diselesaikan secara numerik dan simbolis.

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

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

```
1.41421356237
```

Ini juga berfungsi untuk ekspresi simbolis. Ambil fungsi berikut.

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

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

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

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

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

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

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

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

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

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

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 memeriksa dengan mengevaluasi polinomial pada hasil sebelumnya.

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

$$\begin{aligned} &0.966715594851 \\ &2 \end{aligned}$$

Memecahkan ekspresi simbolis dalam bentuk simbolis mengembalikan 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 ekspresi.

```
> longest sol()
```

$$-0.6180339887498949 \quad 1.618033988749895$$

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])
```

$$0$$

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

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

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

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

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

dengan a=3.

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

Salah satu cara untuk meneruskan parameter tambahan ke f() adalah dengan menggunakan daftar dengan nama fungsi dan parameter (sebaliknya adalah parameter titik koma).

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

$$2.54116291558$$

Ini juga bekerja dengan ekspresi. Tapi kemudian, elemen daftar bernama harus digunakan. (Lebih lanjut tentang daftar di tutorial tentang sintaks EMT).

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

$$2.54116291558$$

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
```

$$\text{universal set}$$

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

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

```
> $&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])
```

$$\begin{aligned} & [6 < x, x < 8, y < -11] \text{ or } [8 < x, y < -11] \\ & \text{or } [x < 8, 13 < y] \text{ or } [x = y, 13 < y] \text{ or } [8 < x, x < y, 13 < y] \\ & \text{or } [y < x, 13 < y] \end{aligned}$$

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

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

Bahasa Matriks

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]
```

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

Produk matriks dilambangkan dengan titik.

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

$$\begin{matrix} 3 \\ 4 \end{matrix}$$

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

$$[3, 4]$$

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

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

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

```
11  
25
```

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

```
1 0  
0 1
```

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

```
>A.A
```

```
7 10  
15 22
```

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

```
1 4  
9 16
```

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

```
37 54  
81 118
```

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

```
37 54  
81 118
```

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

```
1 1  
1 1
```

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

```
0.333333 0.666667  
0.75 1
```

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

```
-2  
2.5
```

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

```
-2  
2.5
```

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

```
1 0  
0 1
```

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

```
1 0  
0 1
```

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

```
1 4  
9 16
```

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

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

```
9  
16
```

Jika salah satu operan adalah vektor atau skalar, itu diperluas secara alami.

```
>2*A
```

```
2 4  
6 8
```

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

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

```
1 4  
3 8
```

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

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

2	6
6	12

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

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

1	2
1	2

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

1	4
3	8

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

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

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

Sekali lagi, ingat bahwa ini bukan produk matriks!

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

55

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

55

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

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

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

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

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

5

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

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

Dari vektor seperti itu, "bukan nol" memilih elemen bukan nol.

Dalam hal ini, kami mendapatkan indeks semua elemen lebih besar dari 50.

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

[8, 9, 10]

Tentu saja, kita dapat menggunakan vektor indeks ini untuk mendapatkan nilai yang sesuai dalam t.

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

[64, 81, 100]

Sebagai contoh, mari kita cari semua kuadrat dari angka 1 hingga 1000, yaitu 5 modulo 11 dan 3 modulo 13.

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

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

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

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

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

112

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

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

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

Ini mengembalikan indeks elemen, yang bukan nol.

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

1	4
2	1
2	2
3	2

Indeks ini dapat digunakan untuk mengatur elemen ke beberapa nilai.

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

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

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

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

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

Dan dimungkinkan untuk mendapatkan elemen dalam vektor.

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

[0.267829, 0.13673, 0.390567, 0.006085]

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

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

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

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

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

[0.765761, 0.952814, 0.548138]

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

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

[0.765761, 0.952814, 0.548138]

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

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

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

Fungsi Matriks Lainnya (Membangun Matriks)

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

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

1	2	3
1	2	3

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

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

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

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

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

```
>A|1
```

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

Dimungkinkan untuk membuat matriks vektor baris dan kolom.

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

1	2	3
1	2	3

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

1	1
2	2
3	3

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

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

1	1
2	4
3	9

Untuk mendapatkan ukuran A, kita dapat menggunakan fungsi berikut.

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

2	1
4	4
[2,	4]
4	

Untuk vektor, ada panjang().

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

9

Ada banyak fungsi lain, yang menghasilkan matriks.

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

1	1
1	1

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

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

[6, 6, 6, 6, 6]

Juga matriks bilangan acak dapat dihasilkan dengan acak (distribusi seragam) atau normal (distribusi Gau).

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

0.66566	0.831835
0.977	0.544258

Berikut adalah fungsi lain yang berguna, yang merestrukturisasi elemen matriks menjadi matriks lain.

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

1	2	3
4	5	6
7	8	9

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

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

Mari kita uji.

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

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

Fungsi multdup() menduplikasi elemen vektor.

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

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

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

Fungsi flipx() dan flipy() mengembalikan urutan baris atau kolom matriks. Yaitu, fungsi flipx() membalik secara horizontal.

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

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

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

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

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

Sebuah fungsi khusus adalah drop(v,i), yang menghilangkan elemen dengan indeks di i dari vektor v.

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

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

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

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

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

```
[0, 5, 0, 6, 0, 0, 0, 7, 0, 8, 0]
```

```
[2, 3, 5, 7, 23, 29, 31, 37, 41, 43, 47]
```

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

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

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

Ada beberapa fungsi khusus untuk mengatur diagonal atau untuk menghasilkan matriks diagonal. Kita mulai dengan matriks identitas.

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

1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1

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

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

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

Perhatikan bahwa kami tidak mengubah matriks A. Kami mendapatkan matriks baru sebagai hasil dari setdiag().

Berikut adalah fungsi, yang mengembalikan matriks tri-diagonal.

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

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

Diagonal suatu matriks juga dapat diekstraksi dari matriks tersebut. Untuk mendemonstrasikan ini, kami merestrukturisasi vektor 1:9 menjadi matriks 3x3.

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

1	2	3
4	5	6
7	8	9

Sekarang kita dapat mengekstrak diagonal.

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

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

Misalnya. 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

Vektorisasi

Hampir semua fungsi di Euler juga berfungsi untuk input matriks dan vektor, kapan pun ini masuk akal. Misalnya, fungsi sqrt() menghitung akar kuadrat dari semua elemen vektor atau matriks.

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

[1, 1.41421, 1.73205]

Jadi Anda dapat dengan mudah membuat tabel nilai. Ini adalah salah satu cara untuk memplot suatu fungsi (alternatifnya menggunakan ekspresi).

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

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

Pada contoh berikut, kita membangkitkan vektor nilai t[i] dengan spasi 0,1 dari -1 hingga 1. Kemudian kita membangkitkan vektor nilai fungsi

$$s = t^3 - t$$

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

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

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

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

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

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

Perhatikan, bahwa ini sangat berbeda dari produk matriks. Produk matriks dilambangkan dengan titik "." di EMT.

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

55

Secara default, vektor baris dicetak dalam format yang ringkas.

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

[1, 2, 3, 4]

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

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

5
25

Untuk mentranspos matriks kita menggunakan apostrof.

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

1
2
3
4

Jadi kita dapat menghitung matriks A kali vektor b.

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

30
70

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

```
>v'.v
```

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

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

```
>v.v'
```

30

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

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

30

Operator dan fungsi mematuhi bahasa matriks Euler.

Berikut ringkasan aturannya.

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

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

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

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

Berikut ini adalah kasus sederhana dengan operator ^.

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

[1, 4, 9]

Berikut adalah kasus yang lebih rumit. Vektor baris dikalikan dengan vektor kolom mengembang keduanya dengan menduplikasi.

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

1	2	3
2	4	6
3	6	9

Perhatikan bahwa produk skalar menggunakan produk matriks, bukan *!

```
>v.v'
```

14

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

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

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

[1, 4, 9]

14

[1, 5, 14]

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

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

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

Untuk menggabungkan matriks dan vektor ada operator "|" dan "_".

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

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

Unsur-unsur matriks disebut dengan "A[i,j]".

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

```
6
```

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

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

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

Indeks juga bisa menjadi vektor baris dari indeks. : menunjukkan semua indeks.

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

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

Bentuk singkat untuk : adalah menghilangkan indeks sepenuhnya.

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

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

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

```
>A{4}
```

```
4
```

Matriks juga dapat diratakan, menggunakan fungsi `redim()`. Ini diimplementasikan dalam fungsi `flatten()`.

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

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

Untuk menggunakan matriks untuk tabel, mari kita reset 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	1	0
0	0	1	0
45	0.785398	0.707107	0.707107
90	1.5708	0	1
135	2.35619	-0.707107	0.707107
180	3.14159	-1	0
225	3.92699	-0.707107	-0.707107
270	4.71239	0	-1
315	5.49779	0.707107	-0.707107
360	6.28319	1	0

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

Dalam contoh berikut, kita menghitung $t[j]^i$ untuk i dari 1 hingga n . Kami mendapatkan matriks, di mana setiap baris adalah tabel t^i untuk satu i . Yaitu, matriks memiliki elemen

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

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

Integrasi numerik `terintegrasi()` hanya berfungsi untuk batas interval skalar. Jadi kita perlu membuat vektor.

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

Kata kunci "peta" membuat vektor fungsi. Fungsinya sekarang akan bekerja untuk vektor bilangan.

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

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

Sub-Matriks dan Matriks-Elemen

Untuk mengakses elemen matriks, gunakan notasi braket.

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

1	2	3
4	5	6
7	8	9
5		

Kita dapat mengakses satu baris matriks yang lengkap.

```
>A[2]
```

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

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

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

```
2
```

Untuk memastikan, Anda mendapatkan baris pertama untuk matriks $1 \times n$ dan $m \times n$, tentukan semua kolom menggunakan indeks kedua kosong.

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

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

Jika indeks adalah vektor indeks, Euler akan mengembalikan baris matriks yang sesuai. Di sini kita ingin baris pertama dan kedua dari A.

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

1	2	3
4	5	6

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

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

7	8	9
4	5	6
1	2	3

Trik indeks bekerja dengan kolom juga.

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

Atau, biarkan indeks pertama kosong.

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

2	3
5	6
8	9

Kita juga bisa mendapatkan baris terakhir dari A.

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

[7, 8, 9]

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

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

4	2	3
4	5	6
7	8	9

Kami juga dapat menetapkan nilai ke baris A.

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

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

Kami bahkan dapat menetapkan 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 mengembalikan matriks kosong, atau pesan kesalahan, tergantung pada pengaturan sistem. Standarnya adalah pesan kesalahan. Ingat, bagaimanapun, bahwa indeks negatif dapat digunakan untuk mengakses elemen matriks yang dihitung dari akhir.

```
>A[4]
```

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

Menyortir dan Mengacak

Fungsi sort() mengurutkan vektor baris.

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

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

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

Mari kita mengocok vektor.

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

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

Indeks berisi urutan yang tepat dari v.

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

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

Ini bekerja untuk vektor string juga.

```
>s=[ "a", "d", "e", "a", "aa", "e"]
```

```
a  
d  
e  
a  
aa  
e
```

```
>{ss,ind}=sort(s); ss
```

```
a  
a  
aa  
d  
e  
e
```

Seperti yang Anda lihat, posisi entri ganda agak acak.

```
>ind
```

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

Fungsi unik mengembalikan daftar elemen unik vektor yang diurutkan.

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

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

Ini bekerja untuk vektor string juga.

```
>unique(s)
```

```
a  
aa  
d  
e
```

Aljabar linier

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

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

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

```
-4  
4.5
```

Untuk contoh lain, kami membuat matriks 200x200 dan jumlah barisnya. Kemudian kita selesaikan $Ax=b$ menggunakan matriks invers. Kami mengukur kesalahan sebagai deviasi maksimal semua elemen dari 1, yang tentu saja merupakan solusi yang benar.

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

8.790745908981989e-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 matriks ini adalah 0.

```
>det(A)
```

0

Matriks Simbolik

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

```
>A &= [a,1,1;1,a,1;1,1,a]; $A
```

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

```
>$&det(A), $&factor(%)
```

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

```
>$&invert(A) with a=0
```

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

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

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

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

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

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

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

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

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

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

Untuk mengekstrak vektor eigen tertentu perlu pengindeksan yang cermat.

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

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

$$[1, -1]$$

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

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

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

Dalam ekspresi simbolik, gunakan dengan.

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

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

Akses ke baris matriks simbolik bekerja seperti halnya dengan matriks numerik.

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

$$[1, a]$$

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

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

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

Ada fungsi simbolik di Maxima untuk membuat vektor dan matriks. Untuk ini, lihat dokumentasi Maxima atau tutorial tentang Maxima di EMT.

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

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

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

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

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

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

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

$$\begin{matrix} -2 & 1 \\ 1.5 & -0.5 \end{matrix}$$

Euler juga memiliki fungsi xinv() yang kuat, yang membuat upaya 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 bisa menggunakan di sini.

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

$$\begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}$$

Misalnya. nilai eigen dari A dapat dihitung secara numerik.

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

```
[16.1168, -1.11684, 0]
```

Atau secara simbolis. Lihat tutorial tentang Maxima untuk detailnya.

```
>$&eigenvalues (@A)
```

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

Nilai Numerik dalam Ekspresi simbolis

Ekspresi simbolis hanyalah 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 simbolik. Saat mentransfer matriks ke bentuk simbolis, pendekatan fraksional untuk real akan digunakan.

```
>$&A
```

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

Untuk menghindarinya, ada fungsi "mxmset(variable)".

```
>m xmset (A); $&A
```

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

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

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

```
1.414213562373095
```

Ketepatan angka floating point besar dapat diubah.

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

```
3.14159265358979323846264338327950288419716939937510582097494\  
4592307816406286208998628034825342117068b0
```

Variabel numerik dapat digunakan dalam ekspresi simbolis apa pun menggunakan "@var". Perhatikan bahwa ini hanya diperlukan, jika variabel telah didefinisikan dengan ":" atau "=" sebagai variabel numerik.

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

```
-5.42477960769379
```

Demo - Suku Bunga

Di bawah ini, kami menggunakan Euler Math Toolbox (EMT) untuk perhitungan 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 5000 (katakanlah dalam dolar).

```
>K=5000
```

```
5000
```

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

```
>K*1.03
```

```
5150
```

Euler akan memahami sintaks berikut juga.

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

```
5150
```

Tetapi lebih mudah menggunakan faktornya

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

```
1.03
```

```
5150
```

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

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

6719.58189672

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

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

6719.58

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

```
>"Starting from " + K + "$ you get " + round(K*q^10,2) + "$."
```

Starting from 5000\$ you get 6719.58\$.

Bagaimana jika kita ingin mengetahui hasil antara dari tahun 1 sampai tahun 9? Untuk ini, bahasa matriks Euler sangat membantu. Anda tidak harus menulis loop, tetapi cukup masukkan

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

Real 1 x 11 matrix

5000.00 5150.00 5304.50 5463.64 ...

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

```
>short 0:10
```

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

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

```
>short q^(0:10)
```

[1, 1.03, 1.0609, 1.0927, 1.1255, 1.1593, 1.1941, 1.2299,
1.2668, 1.3048, 1.3439]

adalah vektor faktor q^0 sampai q^{10} . Ini dikalikan dengan K, dan kami mendapatkan vektor nilai.

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

Tentu saja, cara realistik 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 dua hasil, dengan dan tanpa pembulatan.

```
>longest oneyear(1234.57), longest 1234.57*q
```

```
1271.61  
1271.6071
```

Sekarang tidak ada rumus sederhana untuk tahun ke- n , dan kita harus mengulang selama bertahun-tahun. Euler memberikan banyak solusi untuk ini.

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

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

```
Real 1 x 11 matrix  
5000.00 5150.00 5304.50 5463.64 ...
```

Kami dapat mencetaknya dengan cara yang ramah, menggunakan format kami dengan tempat desimal tetap.

```
>VKr'
```

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

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

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

```
5150.00  
5000.00 5150.00 5304.50
```

Anehnya, kita juga bisa menggunakan vektor indeks. Ingat 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.

Memecahkan Persamaan

Sekarang kita mengambil 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 uang berkurang. Jelas, jika kita hanya mendapatkan 150 bunga di tahun pertama, tetapi menghapus 200, kita kehilangan uang setiap tahun.

Bagaimana kita bisa menentukan berapa tahun uang itu akan bertahan? Kita harus menulis loop untuk ini. Cara termudah adalah dengan iterasi cukup lama.

```
>VKR=iterate("onepay",5000,50)
```

```
Real 1 x 51 matrix  
5000.00 4950.00 4898.50 4845.45 ...
```

Dengan menggunakan bahasa matriks, kita dapat menentukan nilai negatif pertama dengan cara berikut.

```
>min(nonzeros(VKR<0))
```

48.00

Alasan untuk ini adalah bahwa bukan nol($VKR < 0$) mengembalikan vektor indeks i, di mana $VKR[i] < 0$, dan min menghitung indeks minimal.

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

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

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

-19.83
47.00

Mari kita coba menjawab pertanyaan yang lebih ambigu. Asumsikan kita tahu bahwa nilainya adalah 0 setelah 50 tahun. Apa yang akan menjadi tingkat bunga?

Ini adalah pertanyaan yang hanya bisa dijawab dengan angka. Di bawah ini, kita akan mendapatkan formula yang diperlukan. Kemudian Anda akan melihat bahwa tidak ada formula yang mudah untuk tingkat bunga. Tapi untuk saat ini, kami bertujuan untuk solusi numerik.

Langkah pertama adalah mendefinisikan fungsi yang melakukan iterasi sebanyak n kali. Kami menambahkan semua parameter ke fungsi ini.

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

Iterasinya sama seperti di atas

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

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

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

Mari kita coba tes.

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

-19.83

Sekarang kita bisa menyelesaikan masalah kita.

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

3.15

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

Kita dapat menggunakan fungsi yang sama untuk menyelesaikan pertanyaan berikut: Berapa banyak yang dapat kita keluarkan per tahun sehingga modal awal habis setelah 20 tahun dengan asumsi tingkat bunga 3% per tahun.

```
>solve("f(5000,x,3,20)",-200)
```

-336.08

Perhatikan bahwa Anda tidak dapat menyelesaikan jumlah tahun, karena fungsi kami mengasumsikan n sebagai nilai integer.

Solusi Simbolik untuk Masalah Suku Bunga

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

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

$R + q K$

Kita sekarang dapat mengulangi ini.

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

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

images/Shalih Abdillah_22305144009_ApKom2-103-lar

Kami melihat sebuah pola. Setelah n periode yang kita miliki

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

Rumusnya adalah rumus untuk jumlah geometri, yang diketahui Maxima.

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

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

Ini agak rumit. Jumlahnya dievaluasi dengan bendera "simpsum" untuk menguranginya menjadi hasil bagi. Mari kita membuat fungsi untuk ini.

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

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

Fungsi tersebut melakukan hal yang sama seperti fungsi f kita sebelumnya. Tapi itu lebih efektif.

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

```
-19.82504734650985  
-19.82504734652684
```

Kita sekarang dapat menggunakan untuk menanyakan waktu n. Kapan modal kita habis? Dugaan awal kami adalah 30 tahun.

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

```
20.51
```

Jawaban ini mengatakan bahwa itu akan menjadi negatif setelah 21 tahun.

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

Asumsikan kita mendapatkan pinjaman sebesar K, dan membayar n pembayaran sebesar R (dimulai setelah tahun pertama) meninggalkan sisa hutang sebesar Kn (pada saat pembayaran terakhir). Rumus untuk ini jelas

```
> equ &= fs(K, R, P, n)=Kn; $&equ
```

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

Biasanya rumus ini diberikan dalam bentuk

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

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

Kita dapat memecahkan tingkat R secara simbolis.

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

Seperti yang Anda lihat dari rumus, fungsi ini mengembalikan kesalahan titik mengambang untuk $i=0$. Euler tetap merencanakannya.

Tentu saja, kami memiliki batas berikut.

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

Jelas, tanpa bunga kita harus membayar kembali 10 tarif 500.

Persamaan juga dapat diselesaikan untuk n. Kelihatannya lebih bagus, jika kita menerapkan beberapa penye-derhanaan untuk itu.

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

Exercises ** R.2 Integer Exponents

Soal

49

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

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

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

Kita dapat memperoleh hasil tersebut dengan membagi dulu

$$24a^{10}b^{-8}c^7$$

dengan

$$12a^6b^{-3}c^5$$

lalu hasilnya dipangkatkan dengan -5

Soal 50

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

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

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

Hasil dapat diperoleh dengan membagi

$$125p^{12}q^{-14}r^{22}$$

dengan

$$25p^8q^6r^{-15}$$

lalu hasilnya dipangkatkan dengan -4

Soal 88

$$5[3 - 8 \cdot 3^2 + 4 \cdot 6 - 2]$$

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

$$-235$$

Hasil dapat diperoleh dengan mengurangkan 3 dengan hasil perkalian

$$8 * 3^2$$

dan menjumlahkan dengan hasil perkalian

$$4 * 6$$

dan mengurangkan dengan 2. Lalu hasilnya dikali dengan 5

Soal 91

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

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

5

Kita dapat memperoleh hasil tersebut dengan

$$4 * 2^2, 4 * 3, 2 * 8$$

perlu diingat bahwa bilangan berpangkat 0 sama dengan 1 sehingga

$$\frac{(16 - 12 + 16)}{3 + 1}$$

Soal 92

$$\frac{[4(8-6)^2 + 4](3-2.8)}{2^2(2^3+5)}$$

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

-5

Hasil dapat diperoleh dengan melakukan operasi pada setiap tanda kurung, akan dihasilkan

$$\frac{20 * (-13)}{2^2 * 13}$$

lalu ditemukan

$$\frac{-20}{4}$$

R.3 Addition, Subtraction, and Multiplication of Polynomials

Soal

11

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

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

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

Hasil dapat diperoleh dengan mengurangi antar operasi sehingga menghasilkan

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

setelah itu operasikan tiap bilangan dengan bilangan lain yang bervariabel sama

Soal 13

$$(3a^2) (-7a^4)$$

```
> $& (3*a^2) * (-7*a^4)
```

$$-21 a^6$$

Hasil dapat diperoleh dengan mengalikan kedua bilangan

Soal 44

$$(5x + 2y + 3) (5x + 2y - 3)$$

```
> $& (5*x+2*y+3) * (5*x+2*y-3) | ratsimp
```

$$4 y^2 + 20 x y + 25 x^2 - 9$$

Hasil dapat diperoleh dengan mengalikan kedua bilangan, akan ditemukan

$$25x^2 + 10xy - 15x + 10xy + 4y^2 - 6y + 15x + 6y - 9$$

lalu dioperasikan sesuai dengan variabel yang sama

Soal 45

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

```
> $& (x+1) * (x-1) * (x^2+1) | ratsimp
```

$$x^4 - 1$$

Hasil dapat diperoleh dengan mengalihkan

$$(x + 1) dan (x - 1)$$

akan dihasilkan

$$x^2 - 1$$

lalu dikalikan dengan

$$(x^2 + 1)$$

Soal 46

$$(y - 2)(y + 2)(y^2 + 4)$$

```
> $& (y-2)*(y+2)*(y^2+4) | ratsimp
```

$$y^4 - 16$$

Hasil dapat diperoleh dengan mengalikan

$$(y - 2) \text{ dan } (y + 2)$$

akan menghasilkan

$$y^2 - 4$$

lalu dikalikan dengan (y^2+4)

R.4 Factoring

Soal

23

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

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

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

Hasil dapat diperoleh dengan memfaktorkan seperti biasa

Soal 24

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

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

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

Hasil dapat diperoleh dengan memfaktorkan seperti biasa

Soal 47

$$x^2 - 81$$

```
>$& factor(x^2-81)
```

$$(x - 9) (x + 9)$$

Hasil dapat diperoleh dengan memfaktorkan seperti biasa

R.5 The Basics of Equation Solving

Soal

31

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

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

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

Hasil dapat diperoleh dengan mengalikan

$$7(3x + 6)$$

dan mengurangi

$$11 - (x + 2)$$

dan menghasilkan

$$21x + 42 = 11 - x - 2$$

lalu dioperasikan

Soal 37

$$x^2 - 5x = 0$$

```
> $& solve(x^2 - 5*x = 0)
```

$$[x = 0, x = 5]$$

R.6 Rational Expressions

Soal

9

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

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

$$[x = 2]$$

Hasil dapat diperoleh dengan aturan penyebut = 0

$$x^2 - 4x + 4 = 0$$

difaktorkan dan menghasilkan

$$(x - 2)^2 = 0$$

Soal 31

$$\frac{7}{5x} + \frac{3}{5x}$$

```
> $& ratsimp(((7) / (5*x)) + ((3) / (5*x)))
```

$$\frac{2}{x}$$

Hasil dapat diperoleh dengan mengurangi dan menghasilkan

$$\frac{10}{5x}$$

BAB 3

KB PEKAN 4: MENGGUNAKAN EMT UNTUK MENGAMBAR GRAFIK 2 DIMENSI (2D)

article
eumat

Shalih Abdillah(22305144009)

* Menggambar Grafik 2D dengan EMT

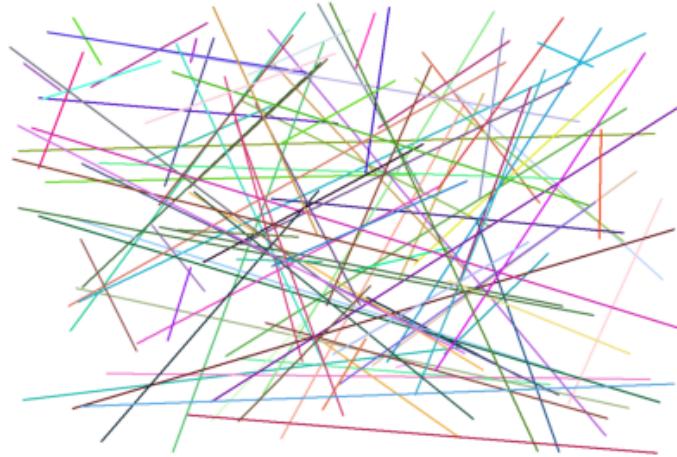
Notebook ini menjelaskan tentang cara menggambar berbagai kurva dan grafik 2D dengan software EMT. EMT menyediakan fungsi plot2d() untuk menggambar berbagai kurva dan grafik dua dimensi (2D).

Plot dasar

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

Dalam contoh, kita hanya menggambar beberapa garis acak dalam berbagai warna. Untuk detail mengenai fungsi-fungsi ini, pelajari fungsi inti EMT.

```
>clc; // membersihkan layar
>window(0,0,1024,1024); // gunakan semua jendela
>setplot(0,1,0,1); // mengatur koordinat plot
>hold on; // mulai mode timpa
>n=100; X=random(n,2); Y=random(n,2); // mendapatkan poin acak
>colors=rgb(random(n),random(n),random(n)); // mendapatkan warna acak
>loop 1 to n; color(colors[#]); plot(X[#],Y[#]); end; // plot
>hold off; // akhiri mode timpa
>insimg; // sisipkan ke notebook
```



```
>reset;
```

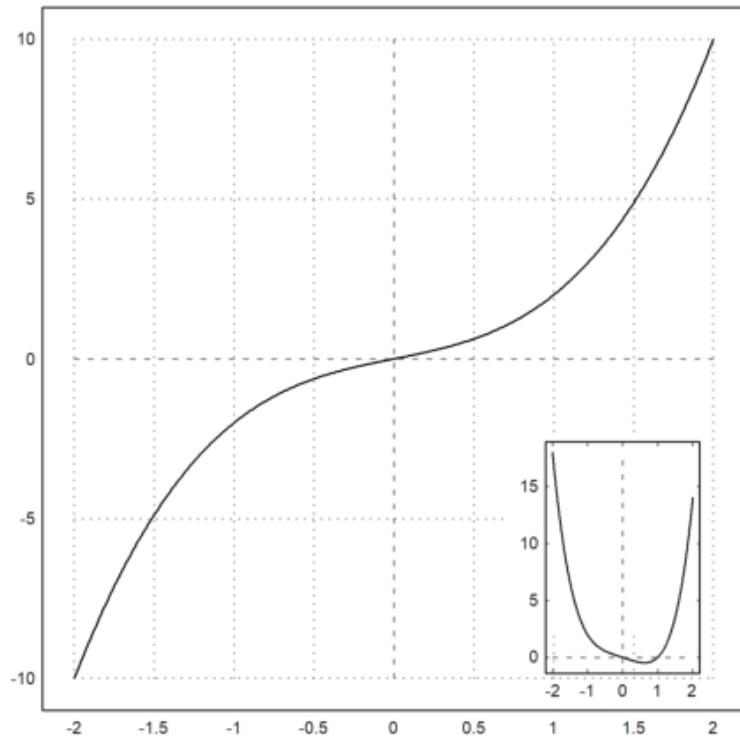
Anda harus menahan grafik, karena perintah `plot()` akan menghapus jendela plot. Untuk menghapus semua yang telah kita lakukan, kita menggunakan `reset()`.

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

Sebagai contoh lain, kita menggambar plot sebagai inset dalam plot lain. Hal ini dilakukan dengan mendefinisikan jendela plot yang lebih kecil.

Perhatikan bahwa jendela ini tidak menyediakan ruang untuk label sumbu di luar jendela plot. Kita harus menambahkan beberapa margin untuk hal ini sesuai kebutuhan. Perhatikan bahwa kita menyimpan dan mengembalikan jendela penuh, dan menahan plot saat ini ketika kita membuat inset.

```
>plot2d("x^3+x");
>xw=750; yw=600; ww=200; hw=300; // ukuran dan posisi grafik kecil
>ow>window();
>window(xw,yw,xw+ww,yw+hw); // agar grafik kedua membuat kolom sendiri
>hold on; // agar perintah grafik atas tidak terhapus
>barclear(xw-50,yw-10,ww+60,ww+60); // ukuran dan posisi grafik besar
>plot2d("x^4-x",grid=6); // grafik besar
```



```
>hold off;
>window(ow);
```

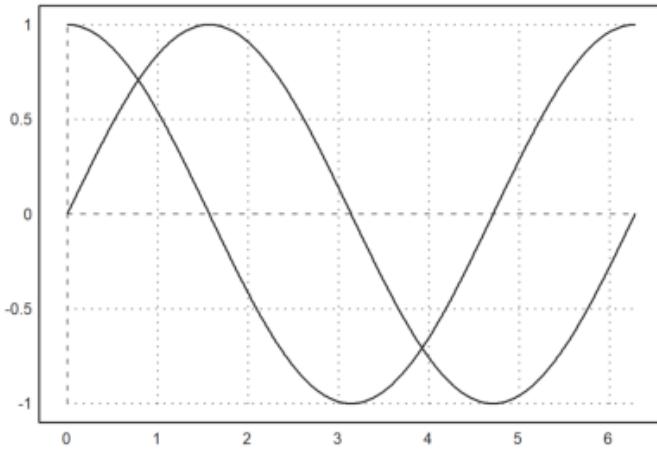
Plot dengan beberapa angka dicapai dengan cara yang sama. Ada fungsi utility figure() untuk ini.

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 grafis saat ini.

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

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



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

Fungsi reset() memulihkan 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.

Hal ini memungkinkan untuk memplot di Maxima menggunakan Gnuplot atau di Python menggunakan Math Plot Lib. Euler dapat memplot plot 2D dari

- ekspresi
- fungsi, variabel, atau kurva yang diparameterkan,
- vektor nilai x-y,
- awan titik-titik di dalam pesawat,
- kurva implisit dengan level atau wilayah level.

- Fungsi yang kompleks

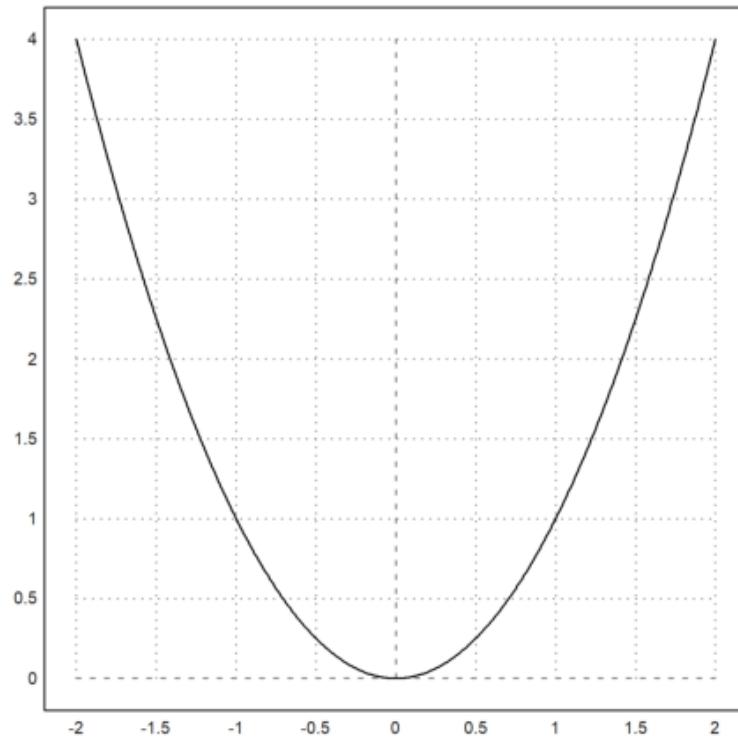
Gaya plot mencakup berbagai gaya untuk garis dan titik, plot batang dan plot berbayang.

Plot Ekspresi atau Variabel

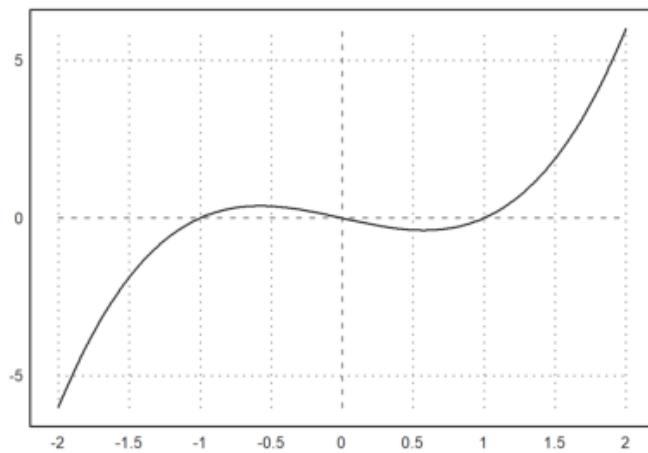
Sebuah ekspresi tunggal dalam "x" (misalnya "4*x^2") atau nama fungsi (misalnya "f") menghasilkan grafik fungsi. Berikut ini adalah contoh paling dasar, yang menggunakan rentang default dan menetapkan rentang y yang tepat agar sesuai dengan plot fungsi.

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

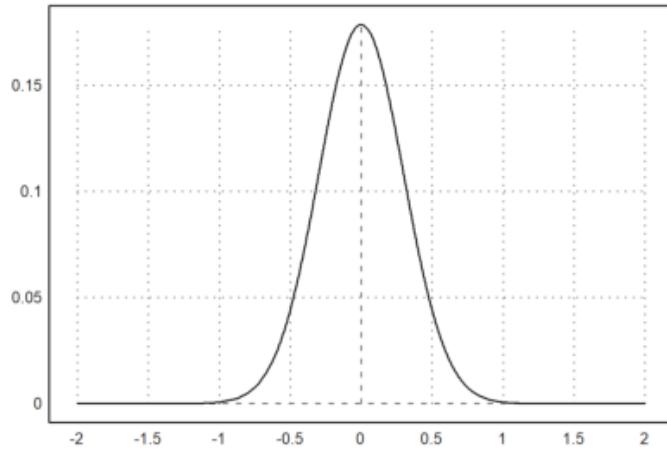
```
>plot2d("x^2");
```



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



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

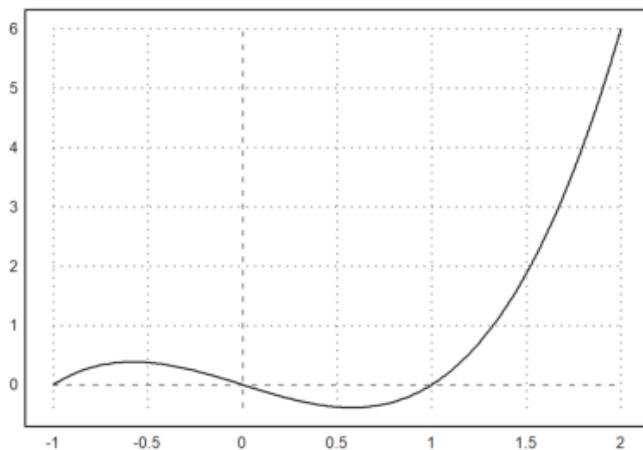


Dari beberapa contoh sebelumnya Anda dapat melihat bahwa aslinya gambar plot menggunakan sumbu X dengan rentang nilai dari -2 sampai dengan 2. Untuk mengubah rentang nilai X dan Y, Anda dapat menambahkan nilai batas X (dan Y) di belakang ekspresi yang digambar.

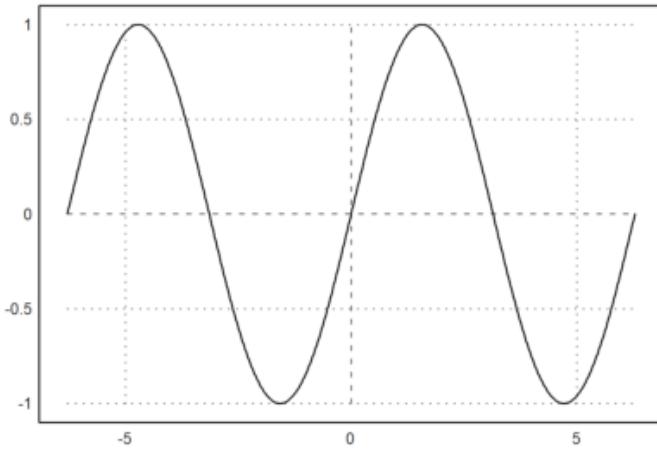
Rentang plot ditetapkan dengan parameter yang ditetapkan berikut ini

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

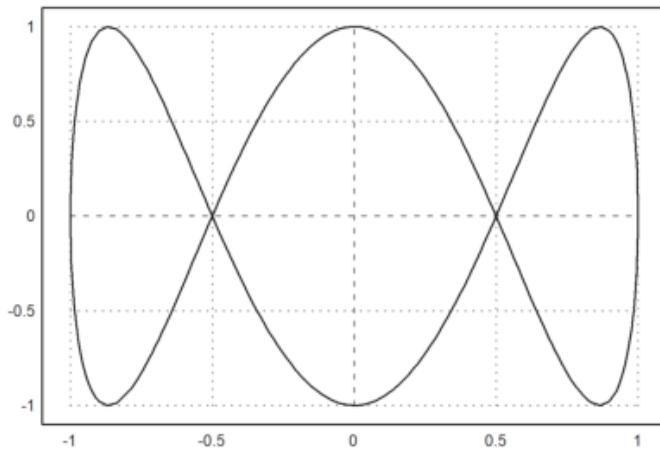
```
>plot2d("x^3-x", -1, 2):
```



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



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



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

Dalam opsi, plot dapat ditetapkan untuk muncul

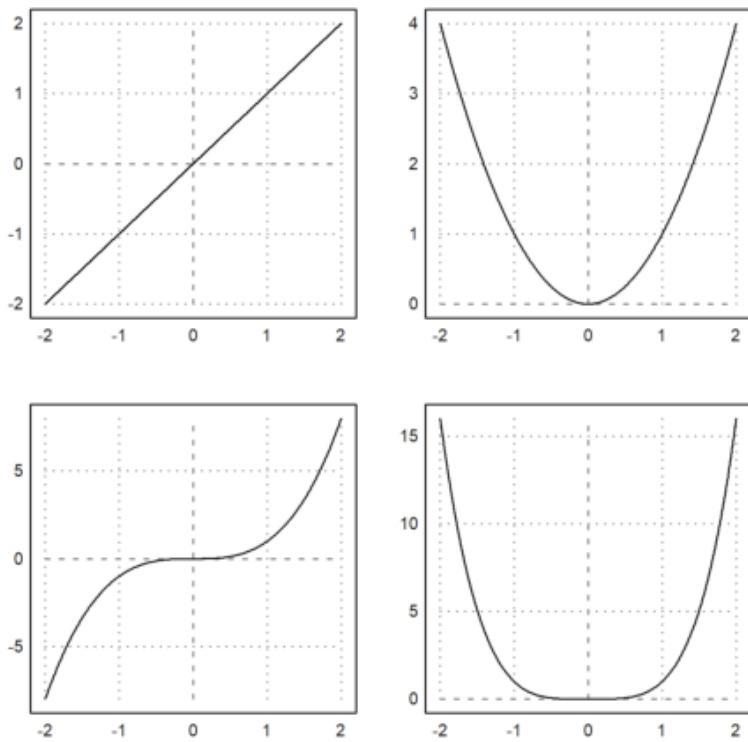
- dalam jendela terpisah yang dapat diubah ukurannya,
- di jendela buku catatan.

Lebih banyak gaya dapat dicapai dengan perintah plot tertentu.

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

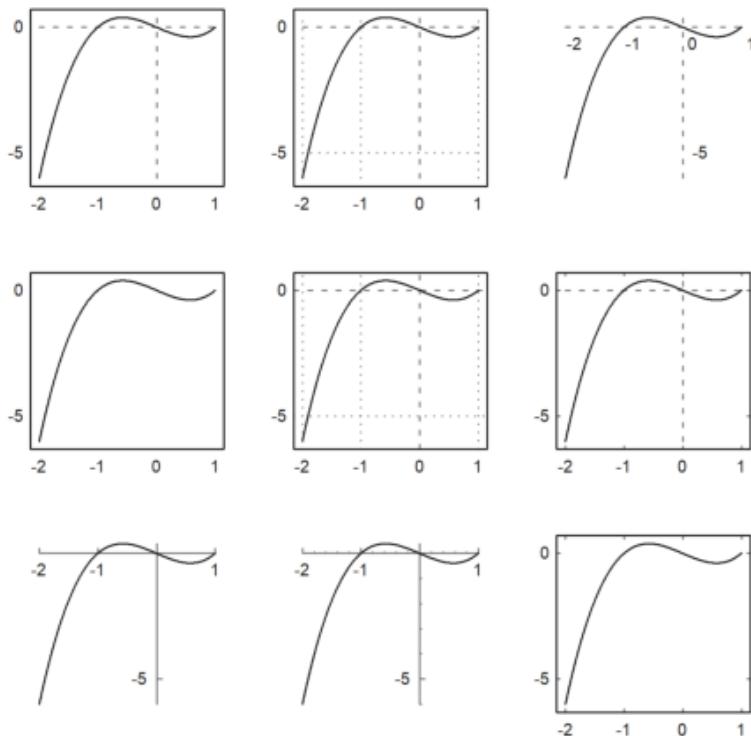
Untuk membagi jendela menjadi beberapa plot, gunakan perintah figure(). Pada contoh, kita memplot x^1 hingga x^4 menjadi 4 bagian jendela. figure(0) mengatur ulang jendela default.

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



In `plot2d()`, there are alternative styles available with `grid=x`. For an overview, we show the various grid styles in one figure (see below for the `figure()` command). The style `grid=0` is not included. It shows no grid and no frame.

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

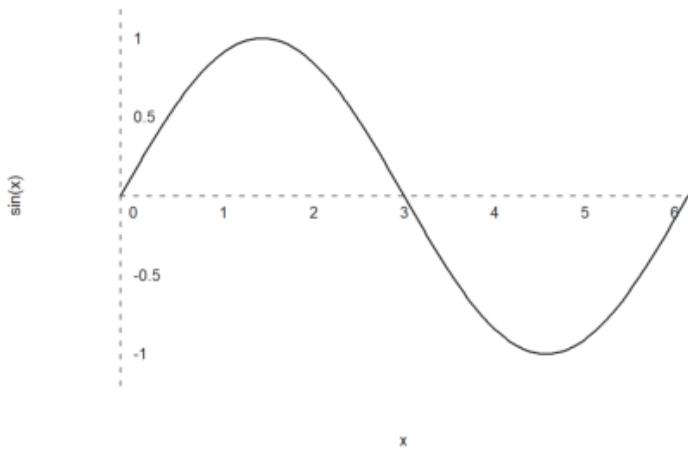


If the arguments to `plot2d()` are an expression followed by four numbers, these numbers are the x- and y-ranges for the plot.

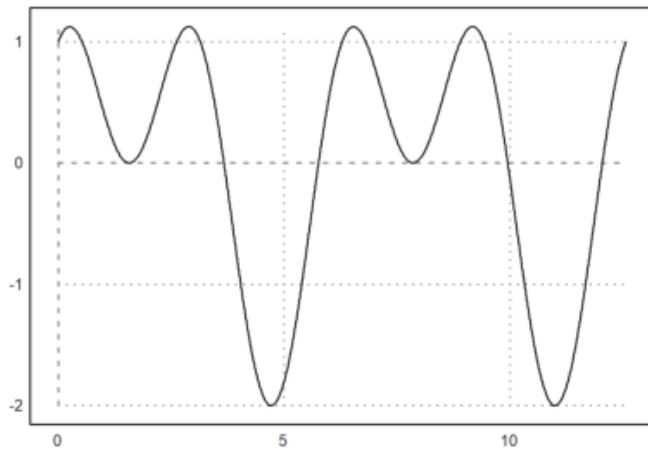
Alternatively, a, b, c, d can be specified as assigned parameters as `a=...` etc.

In the following example, we change the grid style, add labels, and use vertical labels for the y-axis.

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



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

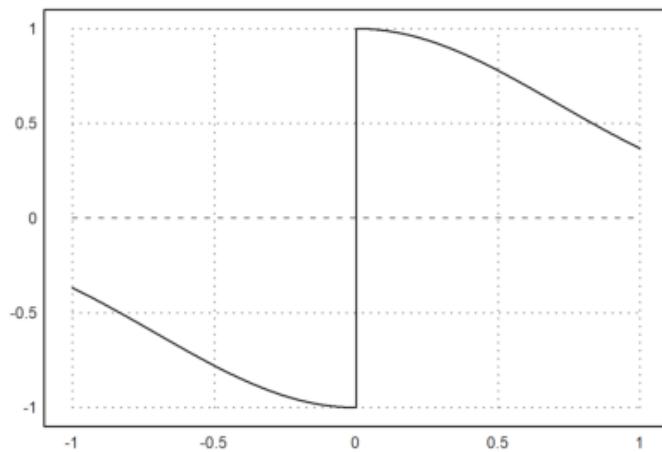


Gambar yang dihasilkan dengan menyisipkan plot ke dalam jendela teks disimpan di direktori yang sama dengan buku catatan, secara default di subdirektori bernama "images". Gambar-gambar ini juga digunakan oleh ekspor HTML.

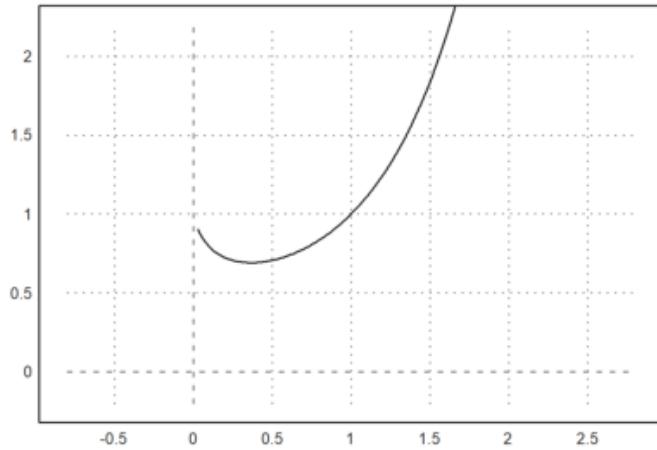
Anda cukup menandai gambar apa pun dan menyalinnya ke clipboard dengan Ctrl-C. Tentu saja, Anda juga dapat mengekspor grafik saat ini dengan fungsi-fungsi dalam menu File.

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

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

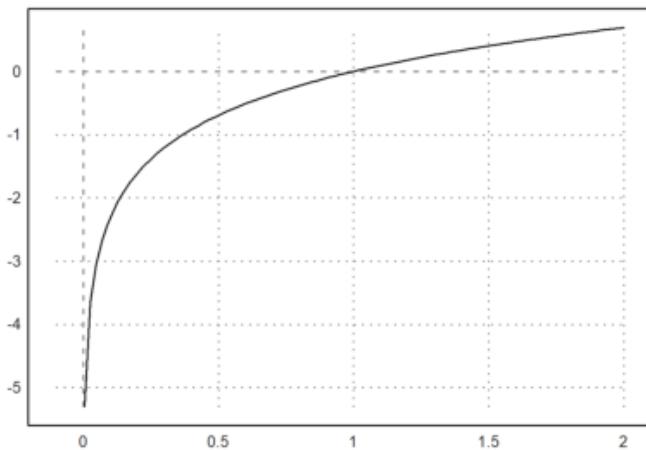


```
>plot2d("x^x", r=1.2, cx=1, cy=1):
```



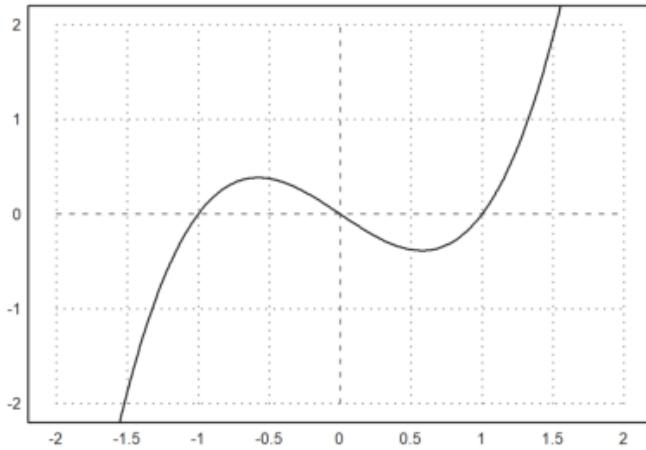
Perhatikan bahwa x^x tidak didefinisikan untuk $x \leq 0$. Fungsi `plot2d` menangkap kesalahan ini, dan mulai memplot segera setelah fungsi didefinisikan. Hal ini berlaku untuk semua fungsi yang mengembalikan NAN di luar jangkauan definisinya.

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

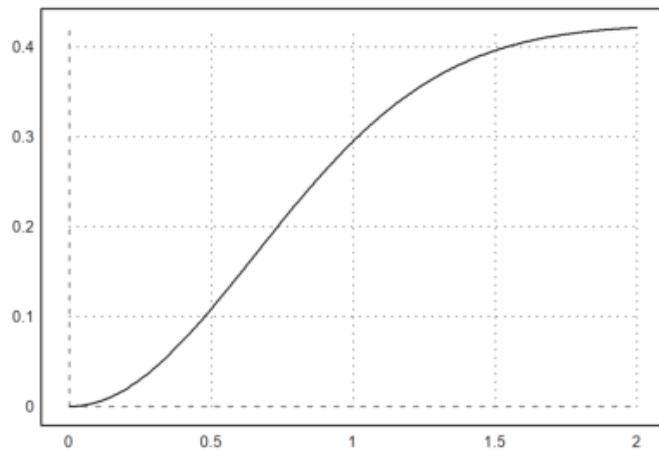


Parameter `square=true` (atau `>square`) memilih rentang y secara otomatis sehingga hasilnya adalah jendela plot persegi. Perhatikan bahwa secara default, Euler menggunakan ruang persegi di dalam jendela plot.

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

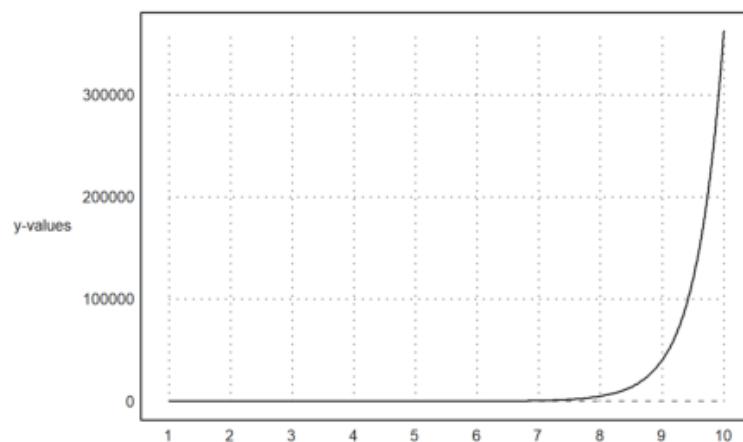


```
>plot2d(''integrate("sin(x)*exp(-x^2)",0,x)'',0,2): // plot integral
```



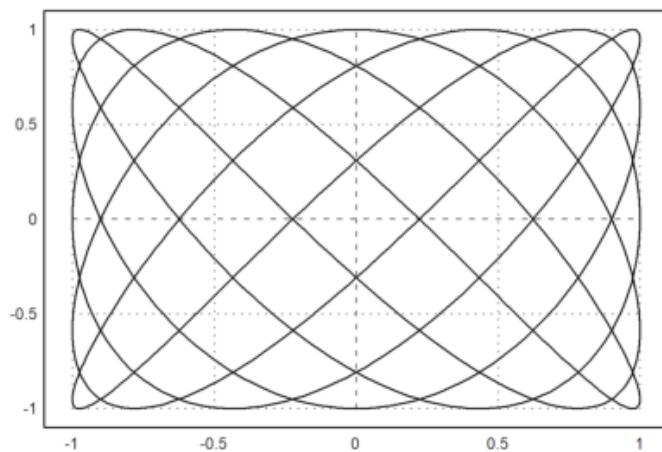
Jika Anda membutuhkan lebih banyak ruang untuk label-y, panggil shrinkwindow() dengan parameter lebih kecil, atau tetapkan nilai positif untuk "lebih kecil" pada plot2d().

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

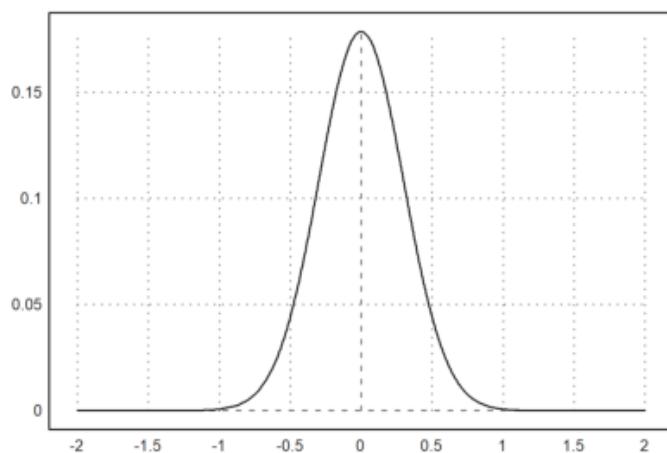


Ekspresi simbolik juga dapat digunakan, karena disimpan sebagai ekspresi string sederhana.

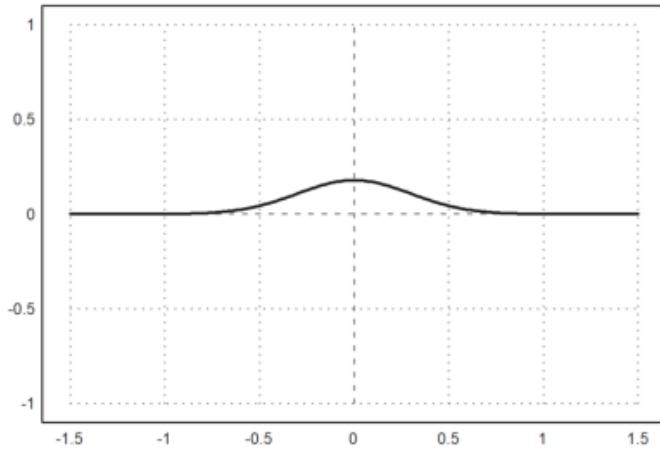
```
>x=linspace(0,2pi,1000); plot2d(sin(5x),cos(7x)):
```



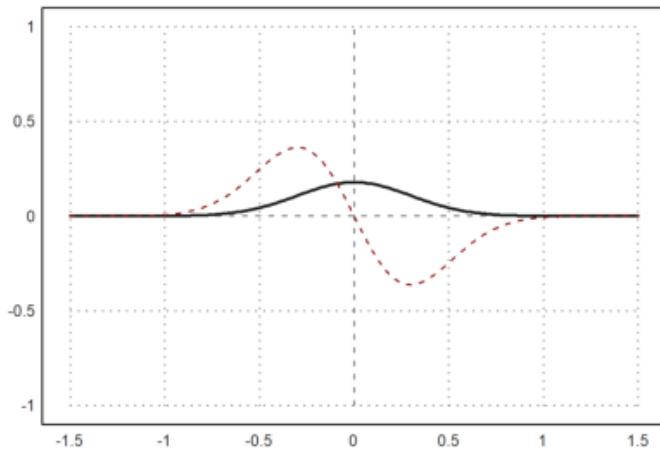
```
>a:=5.6; expr &= exp(-a*x^2)/a; // define expression  
>plot2d(expr,-2,2); // plot from -2 to 2
```



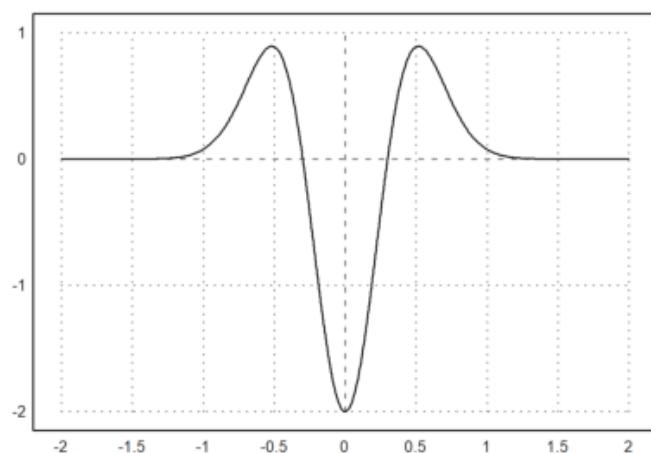
```
>plot2d(expr,r=1,thickness=2); // plot in a square around (0,0)
```



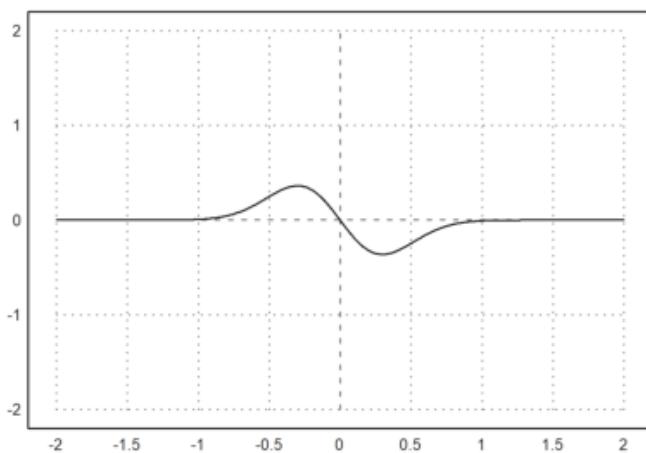
```
>plot2d(&diff(expr,x),>add,style="--",color=red): // add another plot
```



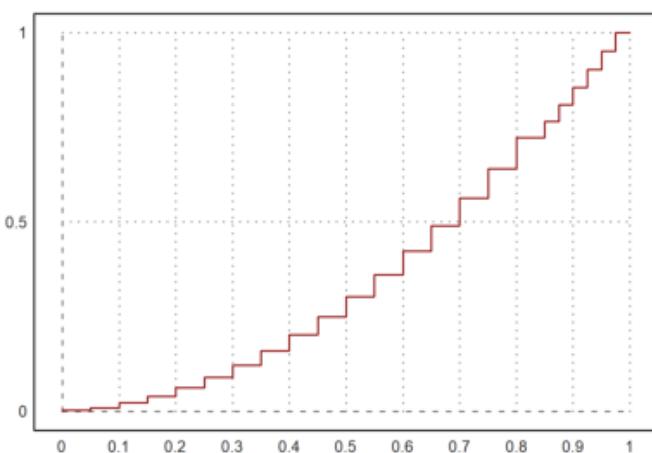
```
>plot2d(&diff(expr,x,2),a=-2,b=2,c=-2,d=1): // plot in rectangle
```



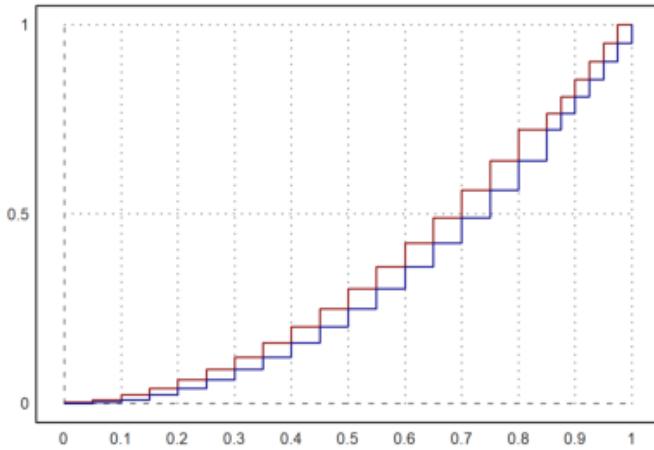
```
>plot2d(&diff(expr,x),a=-2,b=2,>square): // keep plot square
```



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



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

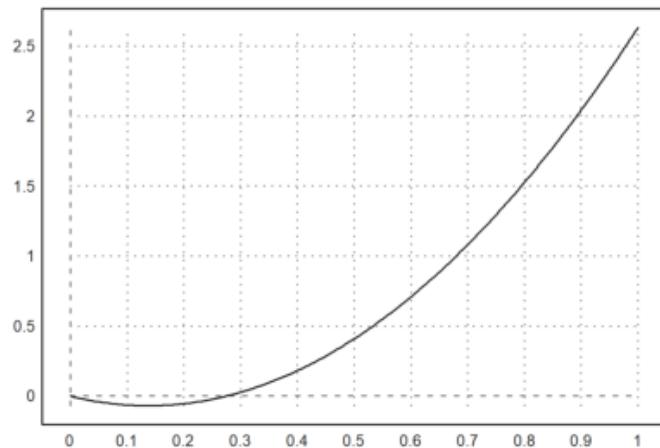


Fungsi dalam satu Parameter

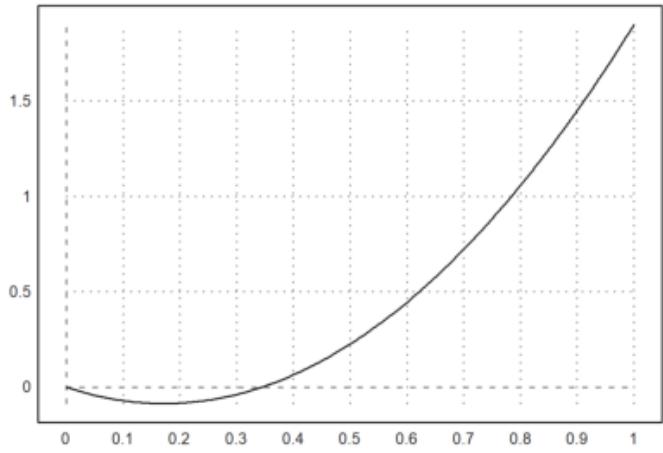
Fungsi plot yang paling penting untuk plot planar adalah `plot2d()`. Fungsi ini diimplementasikan dalam bahasa Euler dalam file "plot.e", yang dimuat pada awal program.

Berikut adalah beberapa contoh penggunaan fungsi. Seperti biasa dalam EMT, fungsi yang bekerja untuk fungsi atau eksekusi lain, Anda dapat mengoper parameter tambahan (selain x) yang bukan variabel global ke fungsi dengan parameter titik koma atau dengan koleksi panggilan.

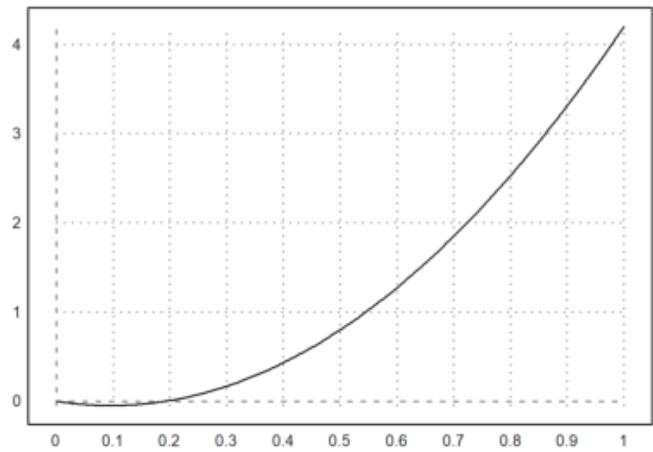
```
>function f(x,a) := x^2/a+a*x^2-x; // define a function
>a=0.3; plot2d("f",0,1;a); // plot with a=0.3
```



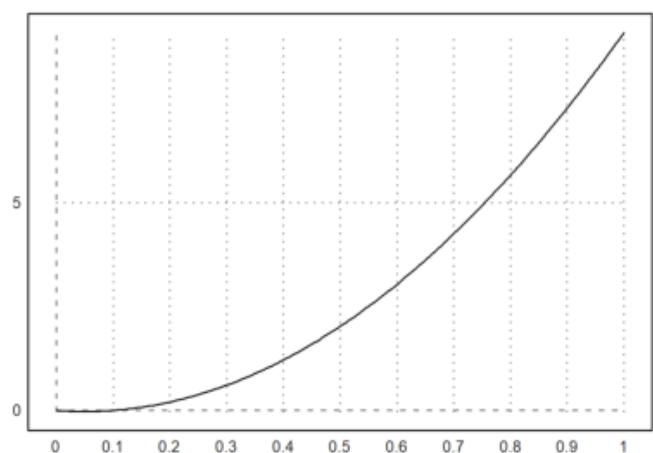
```
>plot2d("f",0,1;0.4); // plot with a=0.4
```



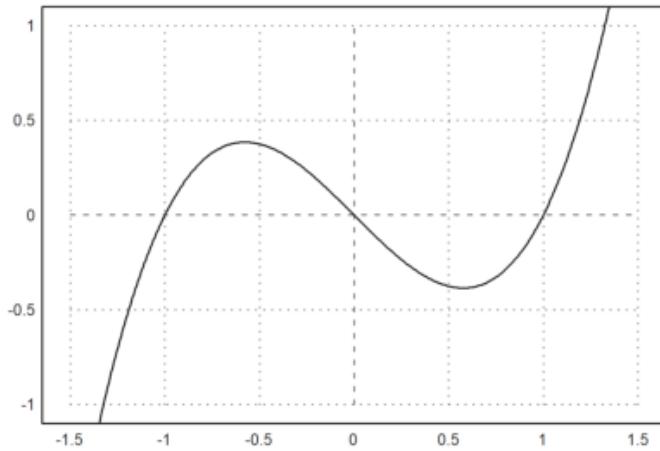
```
>plot2d({{"f",0.2}},0,1); // plot with a=0.2
```



```
>plot2d({{"f(x,b)",b=0.1}},0,1); // plot with 0.1
```



```
>function f(x) := x^3-x; ...
>plot2d("f",r=1):
```



Berikut ini adalah ringkasan dari fungsi yang diterima

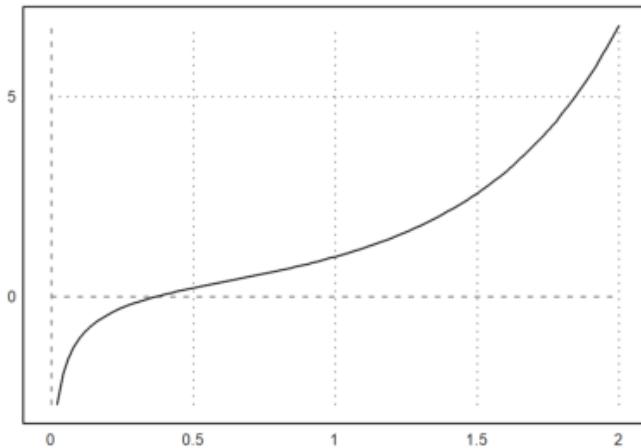
- ekspresi atau ekspresi simbolik dalam x
- fungsi atau fungsi simbolis dengan nama sebagai "f"
- fungsi simbolik hanya dengan nama f

Fungsi plot2d() juga menerima fungsi simbolik. Untuk fungsi simbolik, nama saja sudah cukup.

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

$$x \frac{d}{dx} (x^x) + 1$$

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

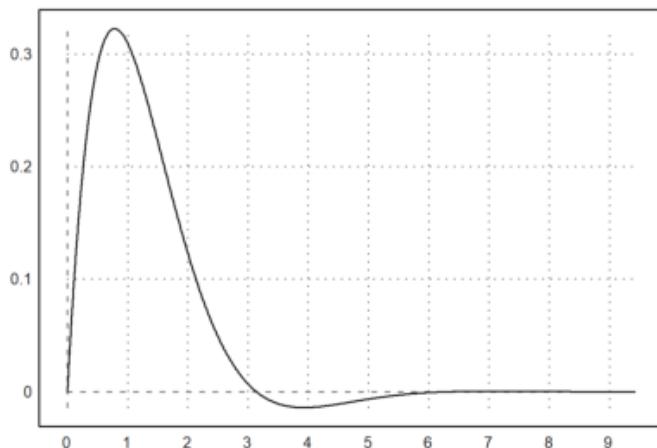


Tentu saja, untuk ekspresi atau ungkapan simbolik, nama variabel sudah cukup untuk memplotnya.

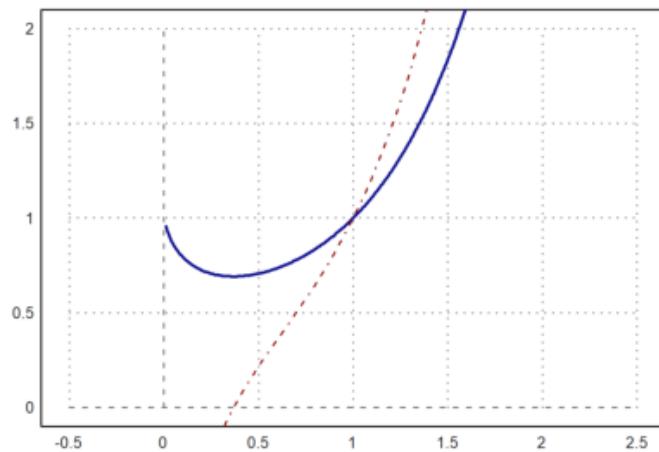
```
>expr &= sin(x)*exp(-x)
```

$$E^{-x} \sin(x)$$

```
>plot2d(expr,0,3pi):
```



```
>function f(x) &= x^x;
>plot2d(f,r=1,cx=1,cy=1,color=blue,thickness=2);
>plot2d(&diff(f(x),x),>add,color=red,style="-.-"):
```



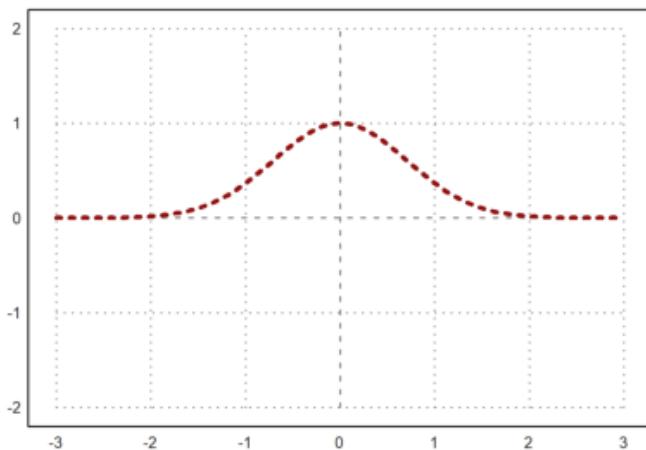
Untuk gaya garis, terdapat berbagai opsi.

- style = "...". Pilih dari "-", "-.", "-.", ".-", "-.-".
- warna: Lihat di bawah untuk warna.
- ketebalan: Standarnya adalah 1.

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

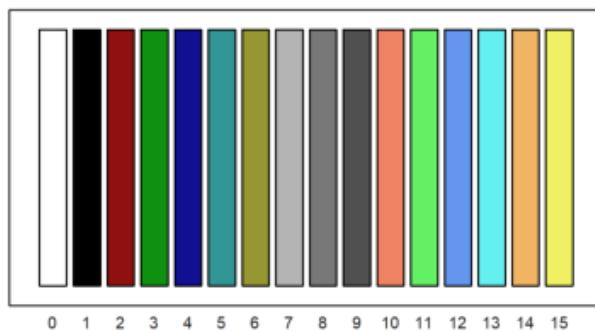
- 0..15: indeks warna default.
- konstanta warna: putih, hitam, merah, hijau, biru, cyan, zaitun, abu-abu muda, abu-abu, abu-abu tua, oranye, hijau muda, biru kehijauan, biru muda, oranye muda, kuning
- rgb (merah, hijau, biru): parameter dalam bentuk real dalam [0,1].

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



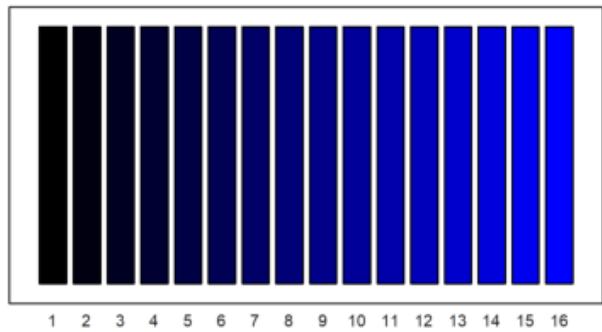
Berikut ini adalah pemandangan warna EMT yang sudah ditetapkan sebelumnya.

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



Tetapi Anda bisa menggunakan warna apa pun.

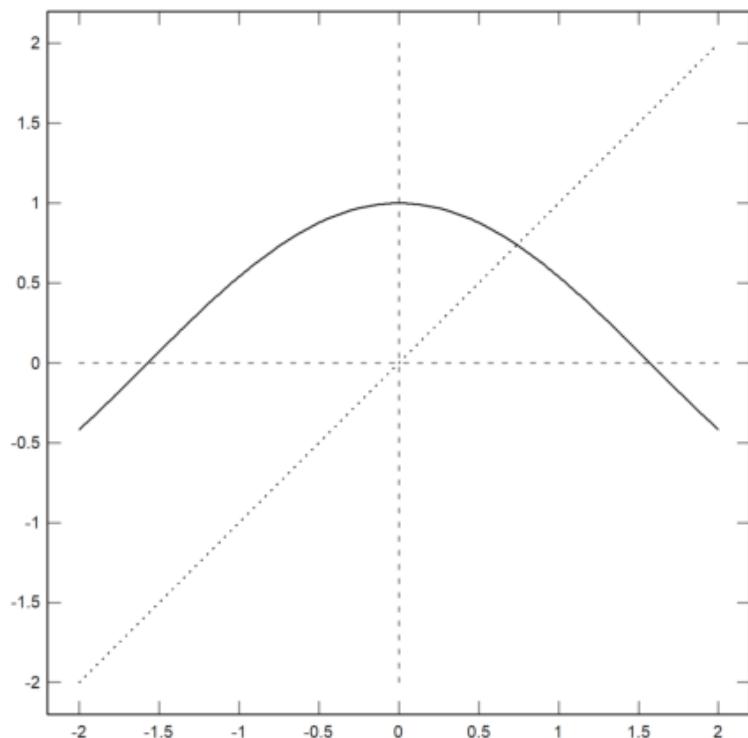
```
>columnsplot(ones(1,16), grid=0, color=rgb(0,0,linspace(0,1,15))) :
```



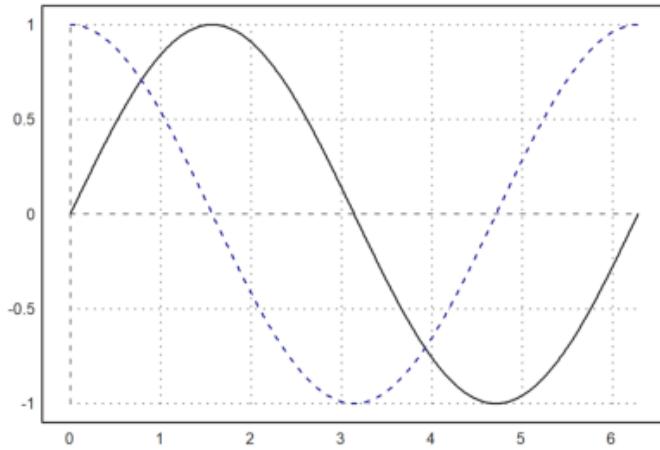
Menggambar Beberapa Kurva pada bidang koordinat yang sama

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):
```

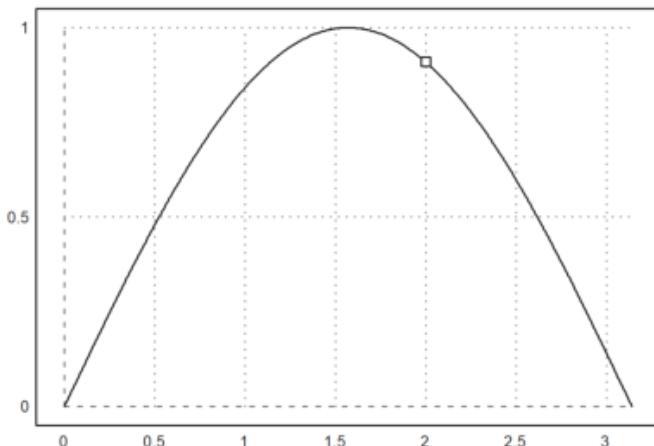


```
>aspect(1.5); plot2d("sin(x)",0,2pi); plot2d("cos(x)",color=blue,style="--",>add):
```



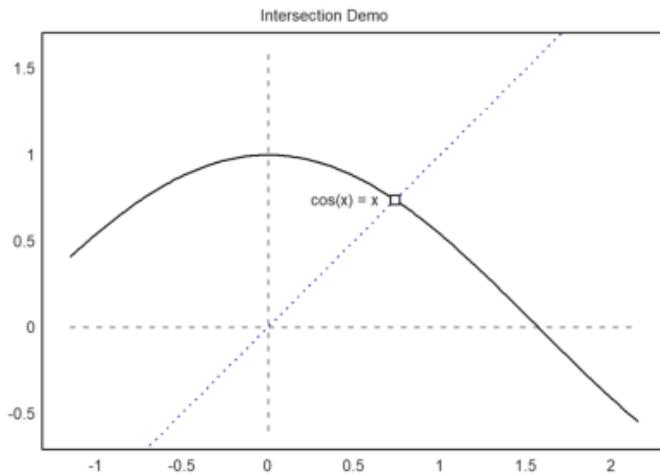
Salah satu kegunaan `>add` adalah untuk menambahkan titik pada kurva.

```
>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 $\text{sinc}(x)=\sin(x)/x$ dan ekspansi Taylor ke-8 dan ke-16. Kami menghitung ekspansi ini menggunakan Maxima melalui ekspresi simbolik.

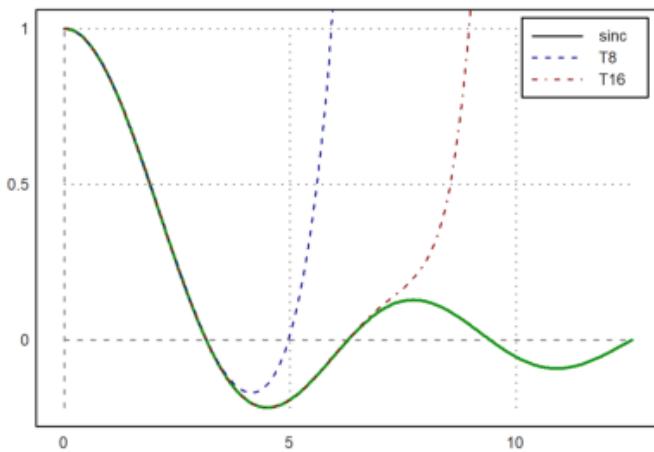
Plot ini dilakukan dalam perintah multi-baris berikut ini dengan tiga kali pemanggilan `plot2d()`. Pemanggilan kedua dan ketiga memiliki set flag `>add`, yang membuat plot menggunakan rentang sebelumnya.

Kami menambahkan kotak label yang menjelaskan fungsinya.

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

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

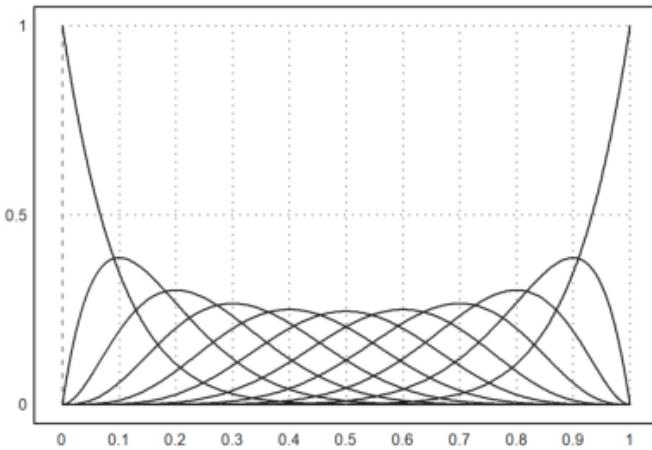
```
>plot2d("sinc(x)",0,4pi,color=green,thickness=2); ...
> plot2d(&taylor(sin(x)/x,x,0,8),>add,color=blue,style="--"); ...
> plot2d(&taylor(sin(x)/x,x,0,16),>add,color=red,style="-.-"); ...
> labelbox(["sinc","T8","T16"],styles=["-","--","-.-"], ...
> colors=[black,blue,red]):
```



Pada contoh berikut, kami menghasilkan Polinomial Bernstein.

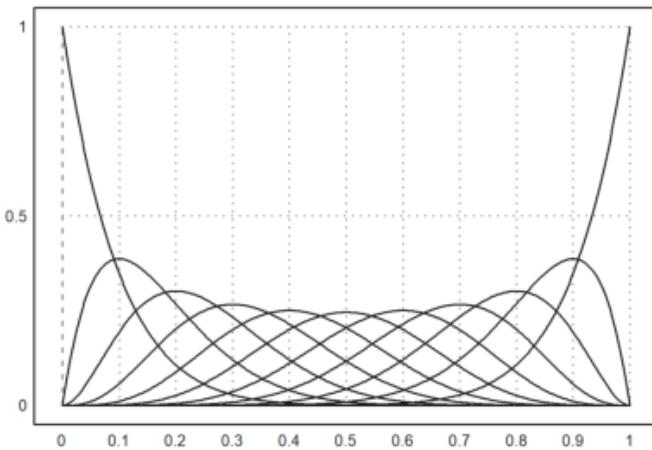
$$B_i(x) = \binom{n}{i} x^i (1-x)^{n-i}$$

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



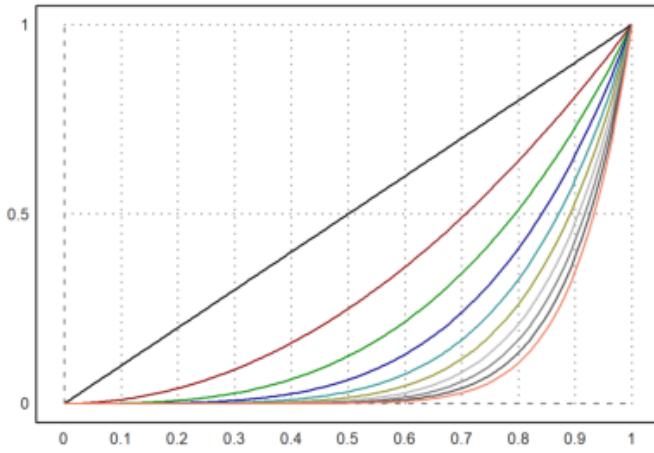
Metode kedua adalah menggunakan sepasang matriks nilai x dan matriks nilai y dengan ukuran yang sama. Kita membuat sebuah matriks nilai dengan satu Bernstein-Polynomial di setiap baris. Untuk ini, kita cukup menggunakan vektor kolom i. Lihatlah pengantar tentang bahasa matriks untuk mempelajari lebih lanjut.

```
>x=linspace(0,1,500);
>n=10; k=(0:n)';
>y=bin(n,k)*x.^k*(1-x).^(n-k); // y is a matrix then
>plot2d(x,y);
```



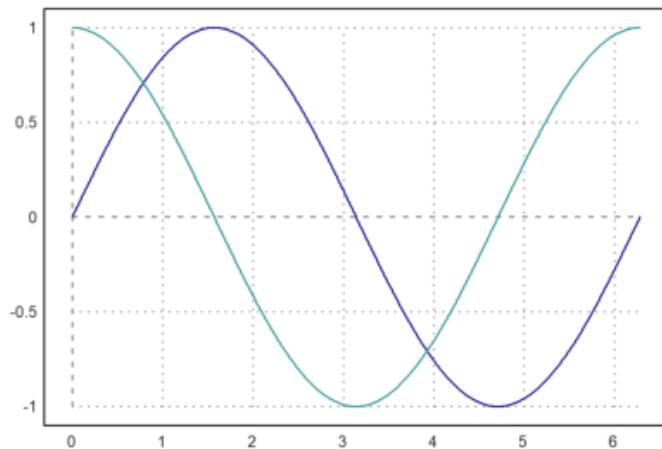
Perhatikan bahwa parameter warna dapat berupa vektor. Kemudian setiap warna digunakan untuk setiap baris matriks.

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

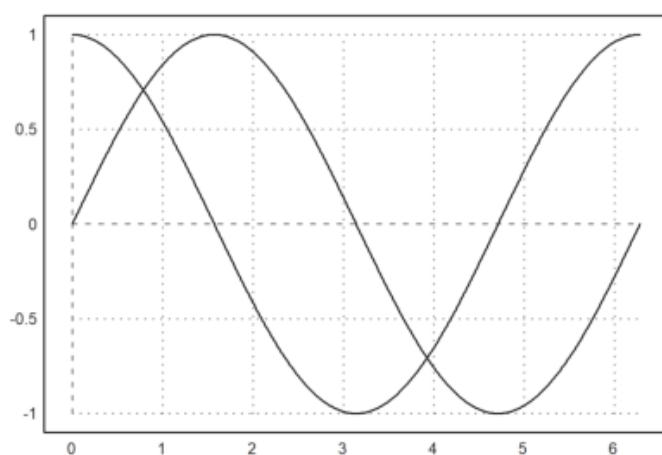


Metode lain adalah menggunakan vektor ekspresi (string). Anda kemudian dapat menggunakan larik warna, larik gaya, dan larik ketebalan dengan panjang yang sama.

```
>plot2d(["sin(x)", "cos(x)"], 0, 2pi, color=4:5):
```



```
>plot2d(["sin(x)", "cos(x)"], 0, 2pi): // plot vector of expressions
```



Kita bisa mendapatkan vektor seperti itu dari Maxima dengan menggunakan makelist() dan mxm2str().

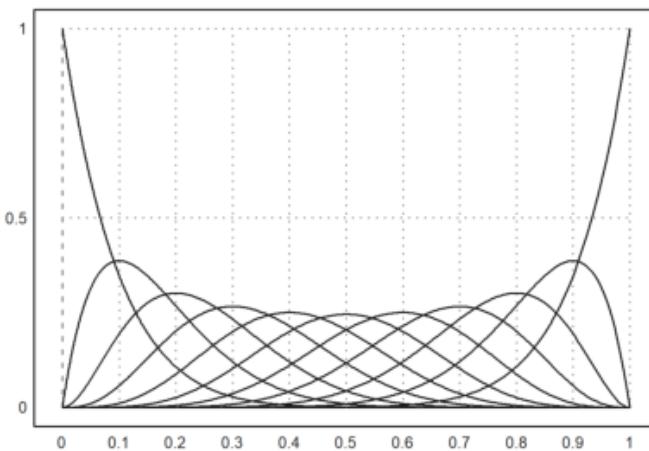
```
>v &= makelist(binomial(10,i)*x^i*(1-x)^(10-i),i,0,10) // make list
```

```
          10          9          8  2          7  3
[(1 - x) , 10 (1 - x) x, 45 (1 - x) x , 120 (1 - x) x ,
 6  4          5  5          4  6          3  7
210 (1 - x) x , 252 (1 - x) x , 210 (1 - x) x , 120 (1 - x) x ,
 2  8          9  10
45 (1 - x) x , 10 (1 - x) x , x ]
```

```
>mxm2str(v) // get a vector of strings from the symbolic vector
```

```
(1-x)^10
10*(1-x)^9*x
45*(1-x)^8*x^2
120*(1-x)^7*x^3
210*(1-x)^6*x^4
252*(1-x)^5*x^5
210*(1-x)^4*x^6
120*(1-x)^3*x^7
45*(1-x)^2*x^8
10*(1-x)*x^9
x^10
```

```
>plot2d(mxm2str(v),0,1): // plot functions
```

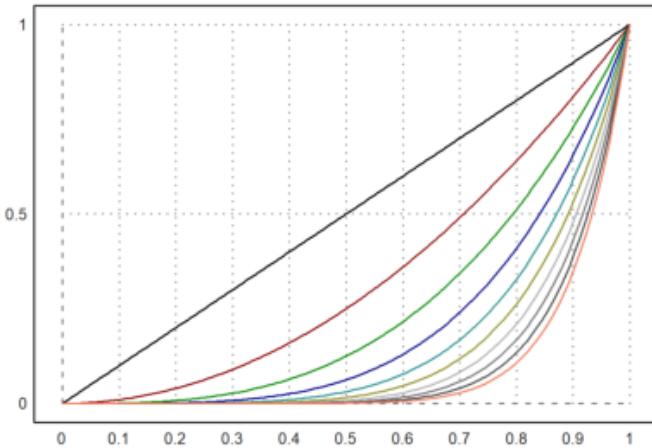


Alternatif lain adalah dengan menggunakan bahasa matriks Euler.

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

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

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

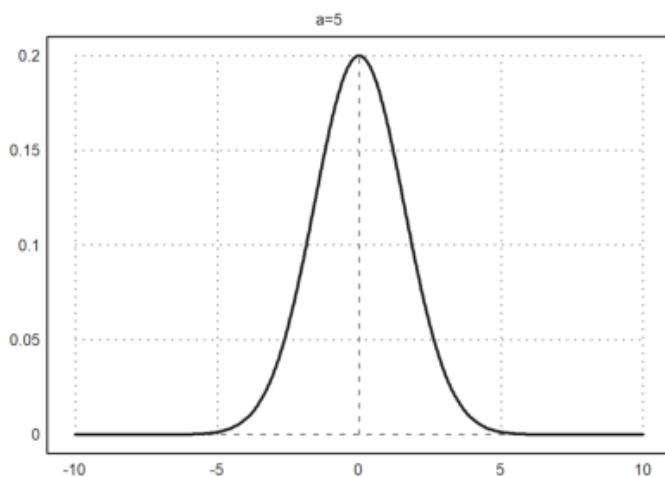


Ekspresi dan fungsi satu baris dapat melihat variabel global.

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

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

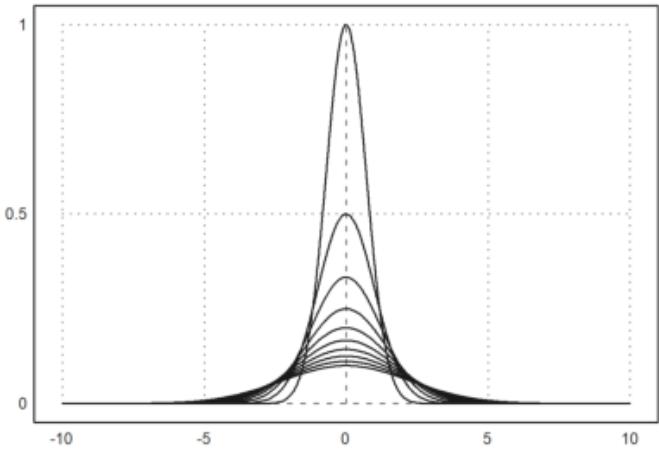
```
>function f(x,a) := 1/a*exp(-x^2/a); ...
>plot2d("f",-10,10;5,thickness=2,title="a=5"):
```



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

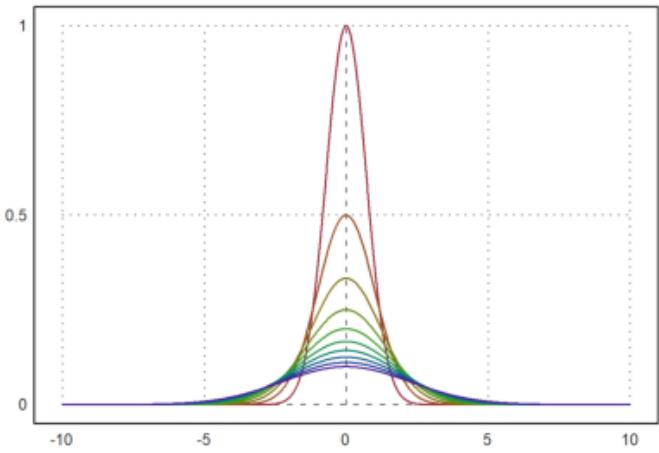
Pada contoh berikut ini, kita menggunakan loop untuk memplot beberapa fungsi (lihat tutorial tentang pemrograman untuk loop).

```
>plot2d({{"f",1}},-10,10); ...
>for a=2:10; plot2d({{"f",a}},>add); end:
```



Kita dapat mencapai hasil yang sama dengan cara berikut menggunakan bahasa matriks EMT. Setiap baris dari matriks $f(x,a)$ adalah satu fungsi. Selain itu, kita dapat mengatur warna untuk setiap baris matriks. Klik dua kali pada fungsi getspectral() untuk penjelasannya.

```
>x=-10:0.01:10; a=(1:10)'; plot2d(x,f(x,a),color=getspectral(a/10)):
```



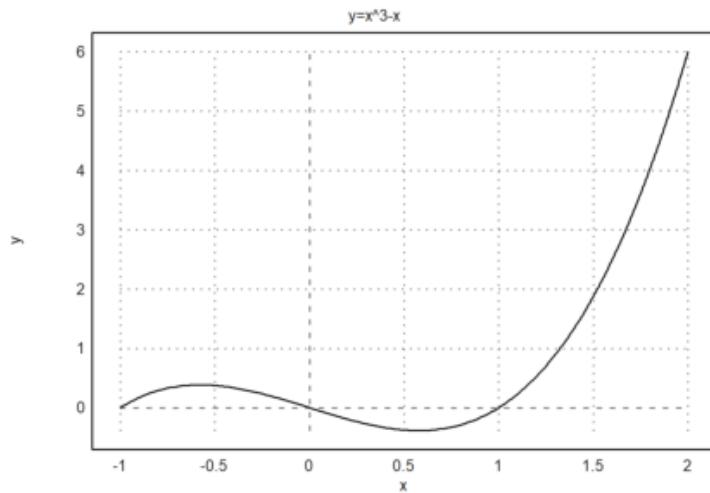
Label Teks

Dekorasi sederhana dapat berupa

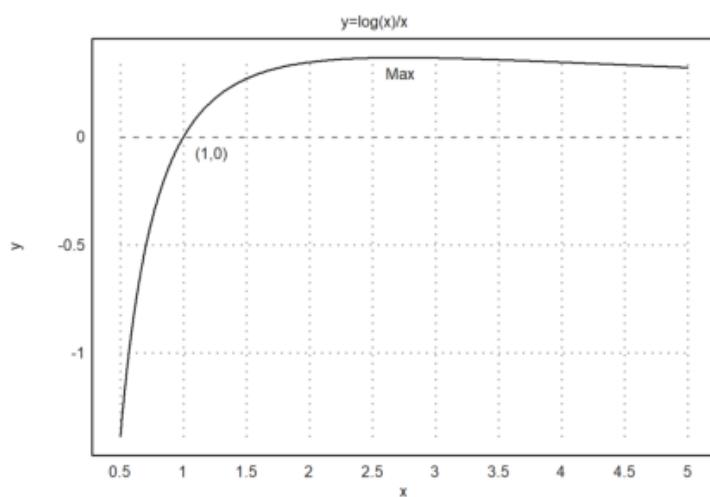
- judul dengan title = "..."
- Label x dan y dengan xl="...", yl="..."
- label teks lain dengan label("...",x,y)

Perintah label akan memplot ke dalam plot saat ini pada koordinat plot (x,y). Perintah ini dapat menerima argumen posisi.

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

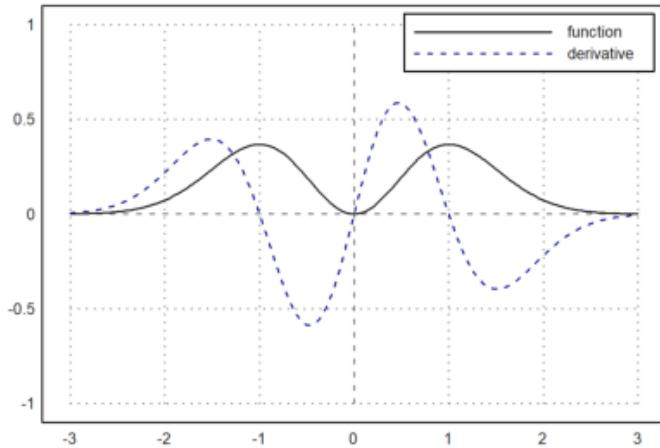


```
>expr := "log(x)/x"; ...
> plot2d(expr,0.5,5,title="y="+expr,xl="x",yl="y"); ...
> label("(1,0)",1,0); label("Max",E,expr(E),pos="lc"):
```



Ada juga fungsi `labelbox()`, yang dapat menampilkan fungsi dan teks. Fungsi ini membutuhkan vektor string dan warna, satu item untuk setiap fungsi.

```
>function f(x) &= x^2*exp(-x^2); ...
>plot2d(&f(x),a=-3,b=3,c=-1,d=1); ...
>plot2d(&diff(f(x),x),>add,color=blue,style="--"); ...
>labelbox(["function","derivative"],styles=[["-", "--"], ...
> colors=[black,blue],w=0.4):
```

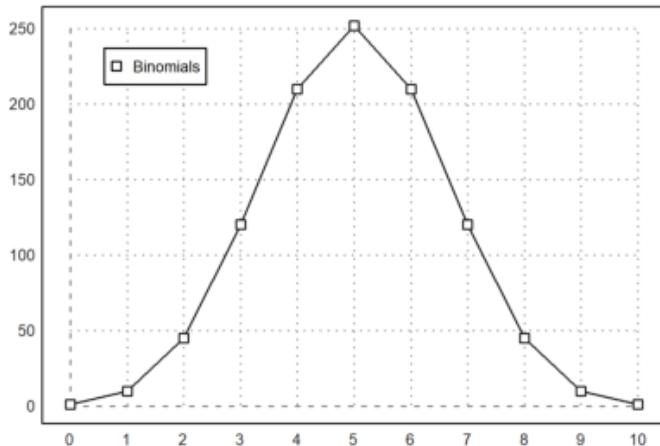


Kotak tersebut berlabuh di kanan atas secara default, tetapi >kiri menambatkannya di kiri atas. Anda dapat memindahkannya ke tempat mana pun yang Anda suka. Posisi jangkar adalah sudut kanan atas kotak, dan angkanya adalah pecahan dari ukuran jendela grafik. Lebarnya adalah otomatis.

Untuk plot titik, kotak label juga dapat digunakan. Tambahkan sebuah parameter >titik, atau sebuah vektor bendera, satu untuk setiap label.

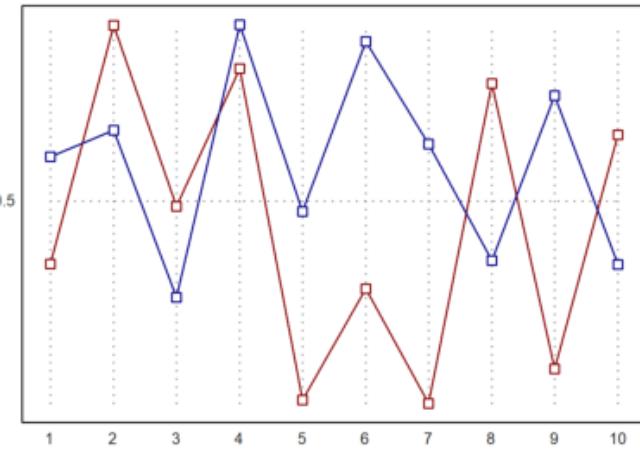
Pada contoh berikut ini, hanya ada satu fungsi. Jadi kita dapat menggunakan string dan bukan vektor string. Kami mengatur warna teks menjadi hitam untuk contoh ini.

```
>n=10; plot2d(0:n,bin(n,0:n),>addpoints); ...
>labelbox("Binomials",styles="[]",>points,x=0.1,y=0.1, ...
>tcolor=black,>left):
```



Gaya plot ini juga tersedia di statplot(). Seperti pada plot2d() warna dapat diatur untuk setiap baris plot. Terdapat lebih banyak plot khusus untuk keperluan statistik (lihat tutorial tentang statistik).

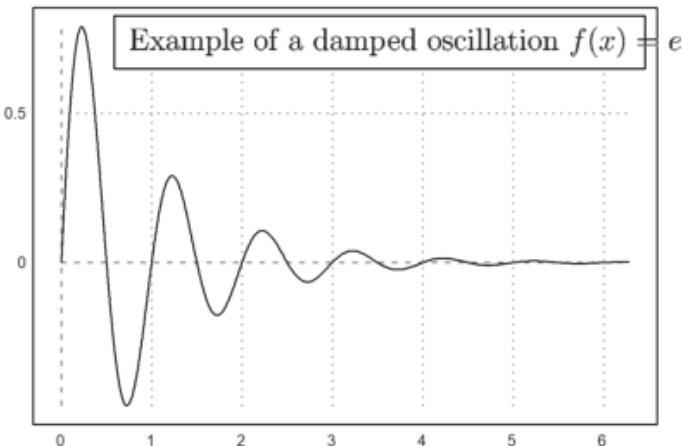
```
>statplot(1:10,random(2,10),color=[red,blue]):
```



Fitur yang serupa adalah fungsi `textbox()`.

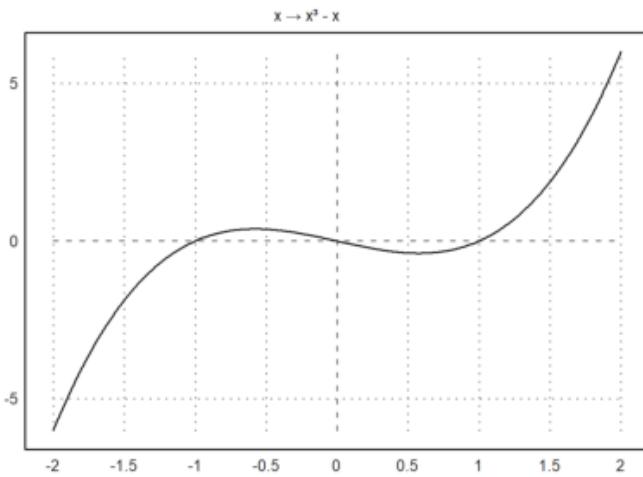
Lebarnya secara default adalah lebar maksimal baris teks. Tetapi, ini juga dapat diatur oleh pengguna.

```
>function f(x) &= exp(-x)*sin(2*pi*x); ...
>plot2d("f(x)",0,2pi); ...
>textbox(latex("\text{Example of a damped oscillation}\\" f(x)=e^{-x}\sin(2\pi x)" ),w=0.85);
```



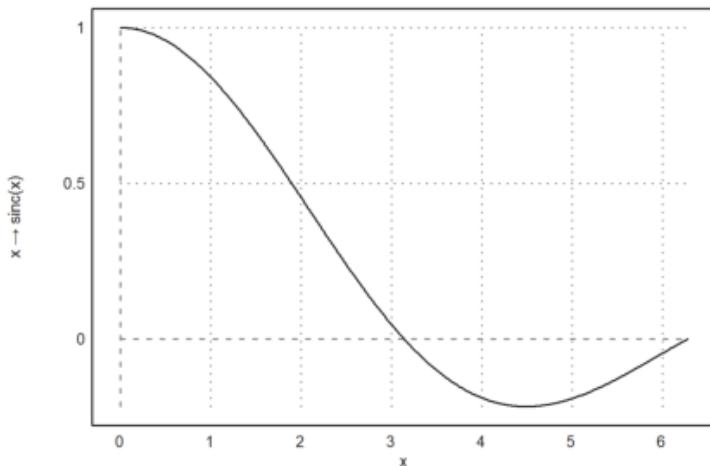
Label teks, judul, kotak label, dan teks lainnya dapat berisi string Unicode (lihat sintaks EMT untuk mengetahui lebih lanjut tentang string Unicode).

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



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

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



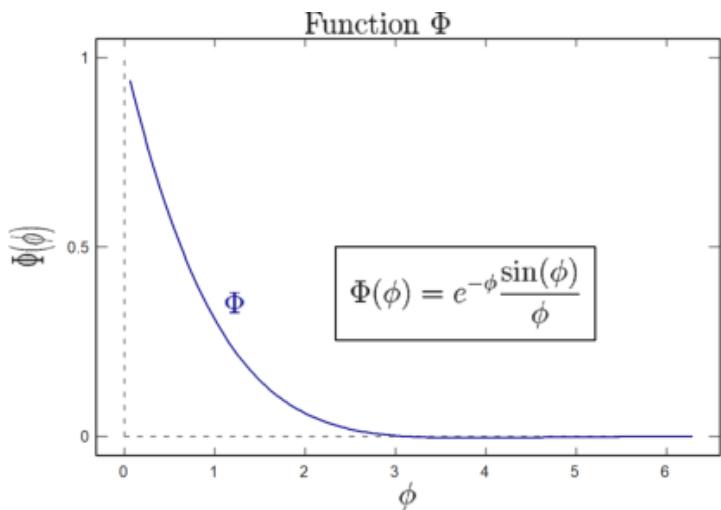
LaTeX

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

Perlu diperhatikan bahwa penguraian LaTeX berjalan lambat. Jika Anda ingin menggunakan LaTeX dalam plot animasi, Anda harus memanggil `latex()` sebelum perulangan sekali dan menggunakan hasilnya (gambar dalam matriks RGB).

Pada plot berikut ini, kita menggunakan LaTeX untuk label x dan y, sebuah label, sebuah kotak label, dan judul plot.

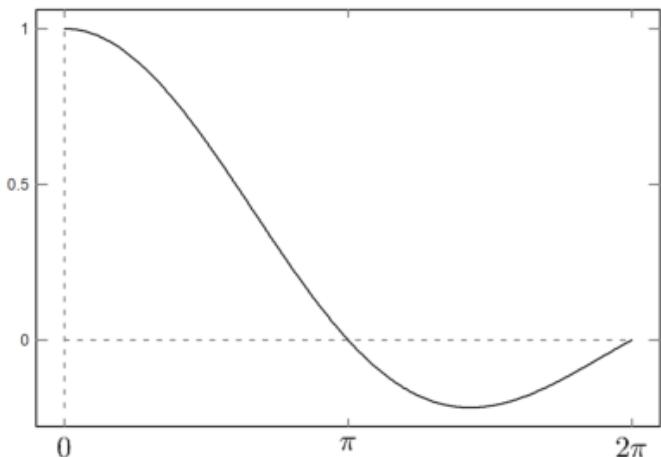
```
>plot2d("exp(-x)*sin(x)/x",a=0,b=2pi,c=0,d=1,grid=6,color=blue, ...
> title=latex("\text{Function } \$\Phi\$"), ...
> xl=latex("\phi"),yl=latex("\Phi(\phi)"); ...
>textbox( ...
> latex("\Phi(\phi) = e^{-\phi} \frac{\sin(\phi)}{\phi}"),x=0.8,y=0.5); ...
>label(latex("\Phi",color=blue),1,0.4):
```



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

Cara termudah adalah dengan membuat plot kosong dengan sebuah frame menggunakan grid=4, dan kemudian menambahkan grid dengan ygrid() dan xgrid(). Pada contoh berikut, kita menggunakan tiga buah string LaTeX untuk label pada sumbu x dengan xtick().

```
>plot2d("sinc(x)",0,2pi,grid=4,<ticks); ...
>ygrid(-2:0.5:2,grid=6); ...
>xgrid([0:2]*pi,<ticks,grid=6); ...
>xlabel([0,pi,2pi],["0","\\pi","2\\pi"],>latex):
```



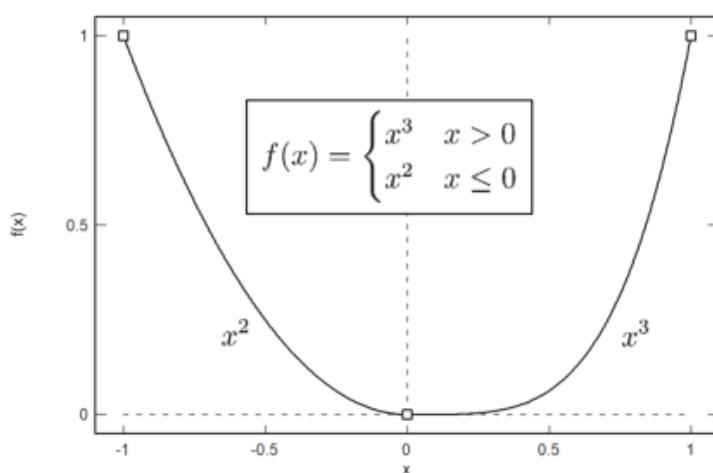
Tentu saja, fungsi juga dapat digunakan.

```
>function map f(x) ...
if x>0 then return x^4
else return x^2
endif
endfunction
```

Parameter "map" membantu menggunakan fungsi untuk vektor. Untuk plot, hal ini tidak diperlukan. Tetapi untuk menunjukkan bahwa vektorisasi berguna, kami menambahkan beberapa titik kunci pada plot pada $x = -1$, $x = 0$ dan $x = 1$.

Pada plot berikut, kita juga memasukkan beberapa kode LaTeX. Kami menggunakannya untuk dua label dan sebuah kotak teks. Tentu saja, Anda hanya dapat menggunakan LaTeX jika Anda telah menginstal LaTeX dengan benar.

```
>plot2d("f",-1,1,xl="x",yl="f(x)",grid=6); ...
>plot2d([-1,0,1],f([-1,0,1]),>points,>add); ...
>label(latex("x^3"),0.72,f(0.72)); ...
>label(latex("x^2"),-0.52,f(-0.52),pos="ll"); ...
>textbox( ...
>  latex("f(x)=\begin{cases} x^3 & x>0 \\ x^2 & x \leq 0 \end{cases}"), ...
>  x=0.7,y=0.2):
```



Interaksi Pengguna

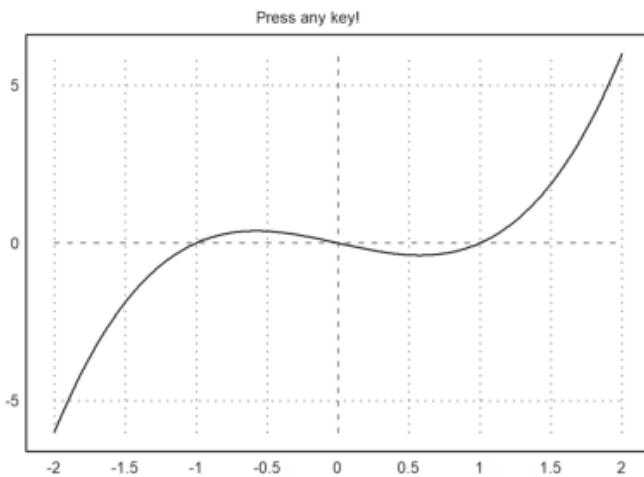
Ketika memplot fungsi atau ekspresi, parameter `>user` memungkinkan pengguna untuk memperbesar dan menggeser plot dengan tombol kurSOR atau mouse. Pengguna dapat

- zoom dengan + atau -
- memindahkan plot dengan tombol kurSOR
- pilih jendela plot dengan mouse
- mengatur ulang tampilan dengan spasi
- keluar dengan kembali

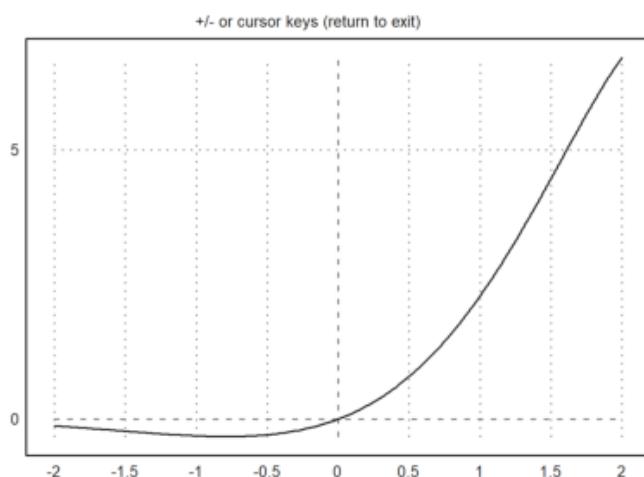
Tombol spasi akan mengatur ulang plot ke jendela plot asli.

Saat memplot data, bendera `>user` hanya akan menunggu penekanan tombol.

```
>plot2d({{"x^3-a*x"},a=1},>user,title="Press any key!":
```



```
>plot2d("exp(x)*sin(x)",user=true, ...
> title="+/- or cursor keys (return to exit)":
```



Berikut ini menunjukkan cara interaksi pengguna tingkat lanjut (lihat tutorial mengenai pemrograman untuk detailnya).

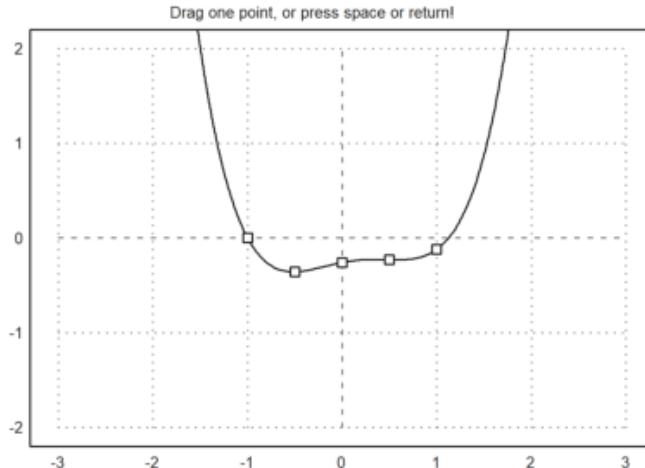
Fungsi bawaan mousedrag() menunggu peristiwa mouse atau keyboard. Fungsi ini melaporkan mouse ke bawah, mouse bergerak atau mouse ke atas, dan penekanan tombol. Fungsi dragpoints() memanfaatkan hal ini, dan mengizinkan pengguna untuk menyeret titik manapun di dalam plot.

Kita membutuhkan fungsi plot terlebih dahulu. Sebagai contoh, kita melakukan interpolasi dalam 5 titik dengan polinomial. Fungsi ini harus memplot ke dalam area plot yang tetap.

```
>function plotf(xp,yp,select) ...
d=interp(xp,yp);
plot2d("interpval(xp,d,x)";d,xp,r=2);
plot2d(xp,yp,>points,>add);
if select>0 then
  plot2d(xp[select],yp[select],color=red,>points,>add);
endif;
title("Drag one point, or press space or return!");
endfunction
```

Perhatikan parameter titik koma pada plot2d (d dan xp), yang diteruskan ke evaluasi fungsi interp(). Tanpa ini, kita harus menulis fungsi plotinterp() terlebih dahulu, untuk mengakses nilai secara global. Sekarang kita menghasilkan beberapa nilai acak, dan membiarkan pengguna menyeret titik-titiknya.

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



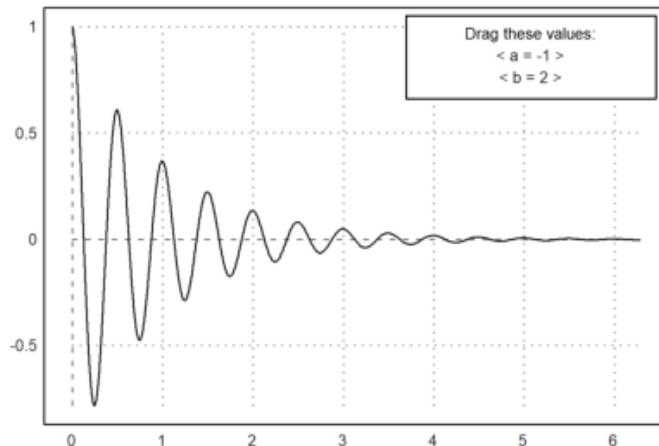
Ada juga fungsi yang memplot fungsi lain tergantung pada vektor parameter, dan memungkinkan pengguna menyesuaikan parameter ini.

Pertama, kita memerlukan fungsi plot.

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

Kemudian kita membutuhkan nama untuk parameter, nilai awal dan matriks rentang nx2, dan secara opsional, sebuah garis judul. Terdapat slider interaktif, yang dapat mengatur nilai oleh pengguna. Fungsi dragvalues() menyediakan ini.

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

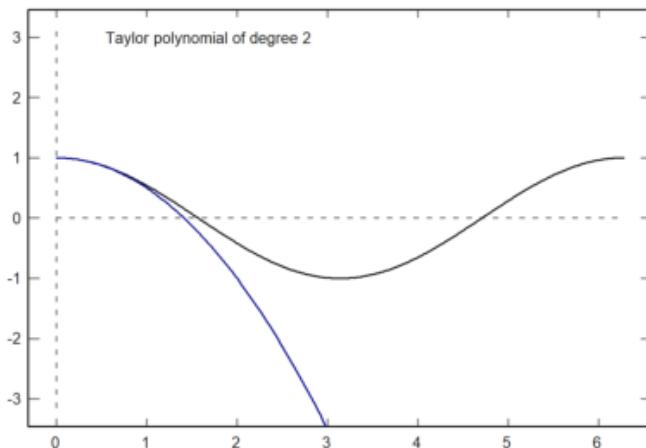


Anda dapat membatasi nilai yang diseret menjadi bilangan bulat. Sebagai contoh, kita menulis fungsi plot, yang memplot polinomial Taylor dengan derajat n ke fungsi kosinus.

```
>function plotf(n) ...
plot2d("cos(x)",0,2pi,>square,grid=6);
plot2d(&"taylor(cos(x),x,0,@n)",color=blue,>add);
textbox("Taylor polynomial of degree "+n,0.1,0.02,style="t",>left);
endfunction
```

Sekarang kita biarkan derajat n bervariasi dari 0 sampai 20 dalam 20 stop. Hasil dari dragvalues() digunakan untuk memplot sketsa dengan n ini, dan untuk menyiapkan plot ke dalam buku catatan

```
>nd=dragvalues("plotf","degree",2,[0,20],20,y=0.8, ...
> heading="Drag the value:"); ...
>plotf(nd);
```



Berikut ini adalah peragaan sederhana dari fungsi ini. Pengguna dapat menggambar di atas jendela plot, meninggalkan jejak titik.

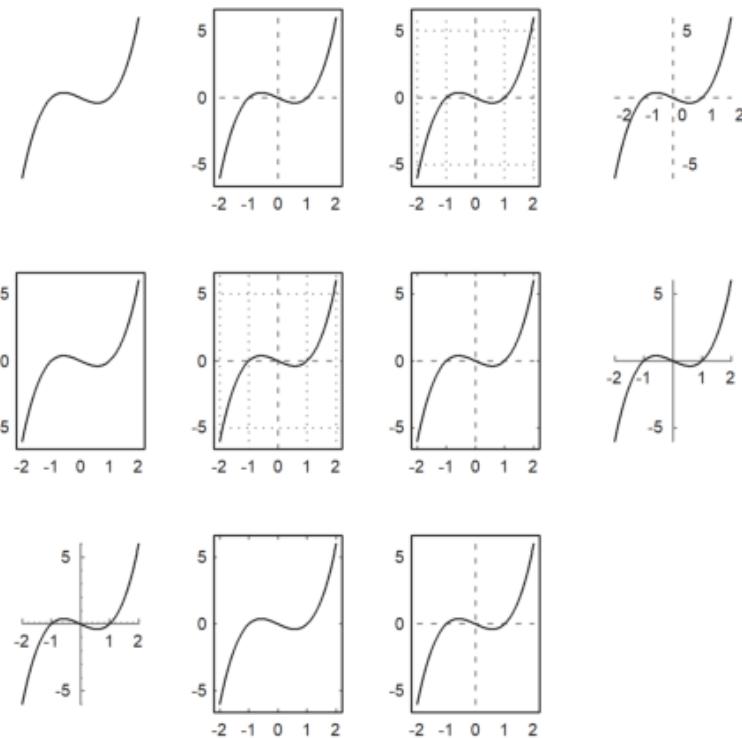
```
>function dragtest ...
plot2d(none,r=1,title="Drag with the mouse, or press any key!");
start=0;
repeat
{flag,m,time}=mousedrag();
if flag==0 then return; endif;
if flag==2 then
    hold on; mark(m[1],m[2]); hold off;
endif;
end
endfunction
```

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

Gaya Plot 2D

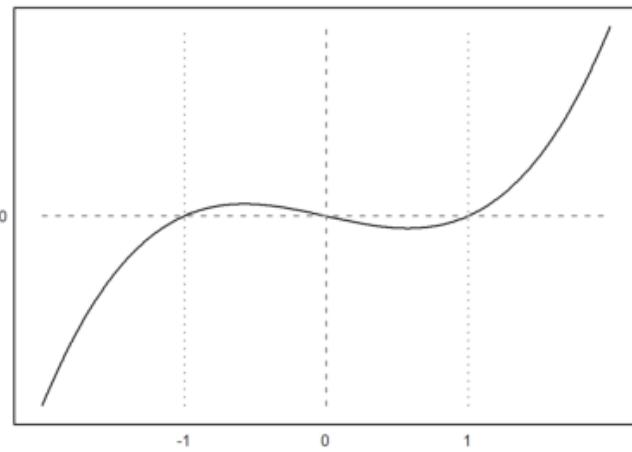
Secara default, EMT menghitung tick sumbu otomatis dan menambahkan label pada setiap tick. Hal ini dapat diubah dengan parameter kisi-kisi. Gaya default sumbu dan label dapat dimodifikasi. Selain itu, label dan judul dapat ditambahkan secara manual. Untuk mengatur ulang ke gaya default, gunakan reset().

```
>aspect();
>figure(3,4); ...
> figure(1); plot2d("x^3-x",grid=0); ... // no grid, frame or axis
> figure(2); plot2d("x^3-x",grid=1); ... // x-y-axis
> figure(3); plot2d("x^3-x",grid=2); ... // default ticks
> figure(4); plot2d("x^3-x",grid=3); ... // x-y- axis with labels inside
> figure(5); plot2d("x^3-x",grid=4); ... // no ticks, only labels
> figure(6); plot2d("x^3-x",grid=5); ... // default, but no margin
> figure(7); plot2d("x^3-x",grid=6); ... // axes only
> figure(8); plot2d("x^3-x",grid=7); ... // axes only, ticks at axis
> figure(9); plot2d("x^3-x",grid=8); ... // axes only, finer ticks at axis
> figure(10); plot2d("x^3-x",grid=9); ... // default, small ticks inside
> figure(11); plot2d("x^3-x",grid=10); ...// no ticks, axes only
> figure(0):
```



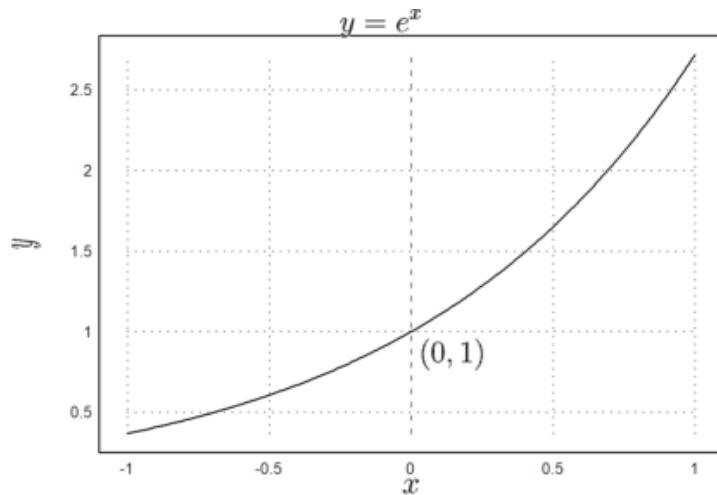
Parameter <frame> mematikan bingkai, dan <framecolor=blue> menetapkan bingkai ke warna biru. Jika Anda menginginkan tanda centang Anda sendiri, Anda dapat menggunakan style=0, dan menambahkan semuanya nanti.

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



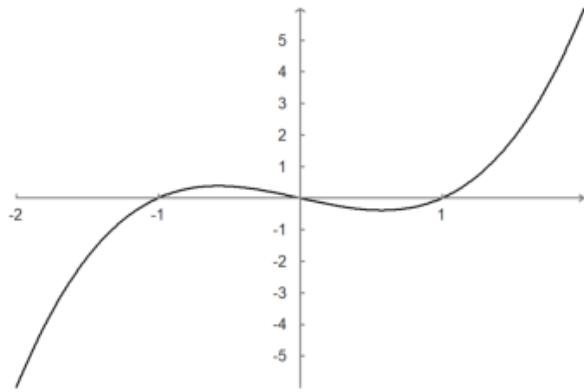
Untuk judul plot dan label sumbu, lihat contoh berikut.

```
>plot2d("exp(x)",-1,1);
>textcolor(black); // set the text color to black
>title(latex("y=e^x")); // title above the plot
>xlabel(latex("x")); // "x" for x-axis
>ylabel(latex("y"),>vertical); // vertical "y" for y-axis
>label(latex("(0,1)'),0,1,color=blue): // label a point
```



Sumbu dapat digambar secara terpisah dengan sumbu x() dan sumbu y().

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

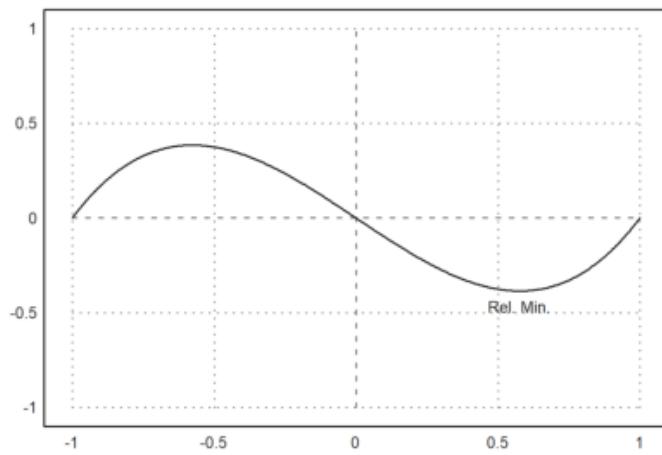


Teks pada plot dapat diatur dengan `label()`. Pada contoh berikut ini, "lc" berarti lower center. Ini mengatur posisi label relatif terhadap koordinat plot.

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

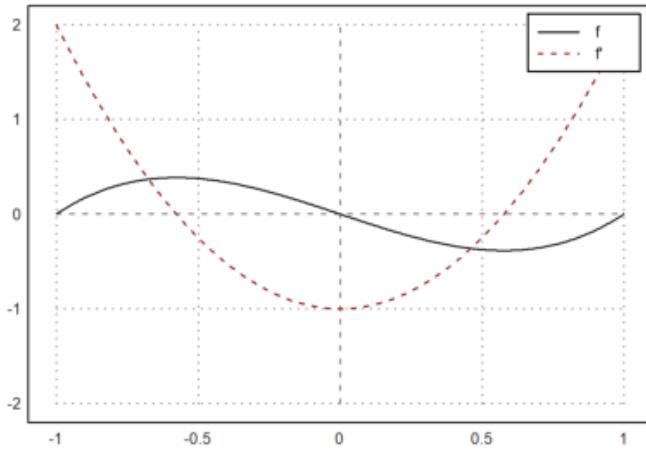
$$x^3 - x$$

```
>plot2d(f,-1,1,>square);
>x0=fmin(f,0,1); // compute point of minimum
>label("Rel. Min.",x0,f(x0),pos="lc"); // add a label there
```

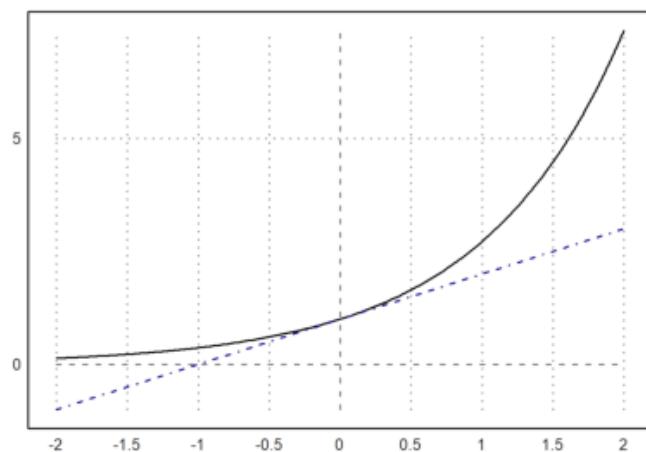


Terdapat juga kotak teks.

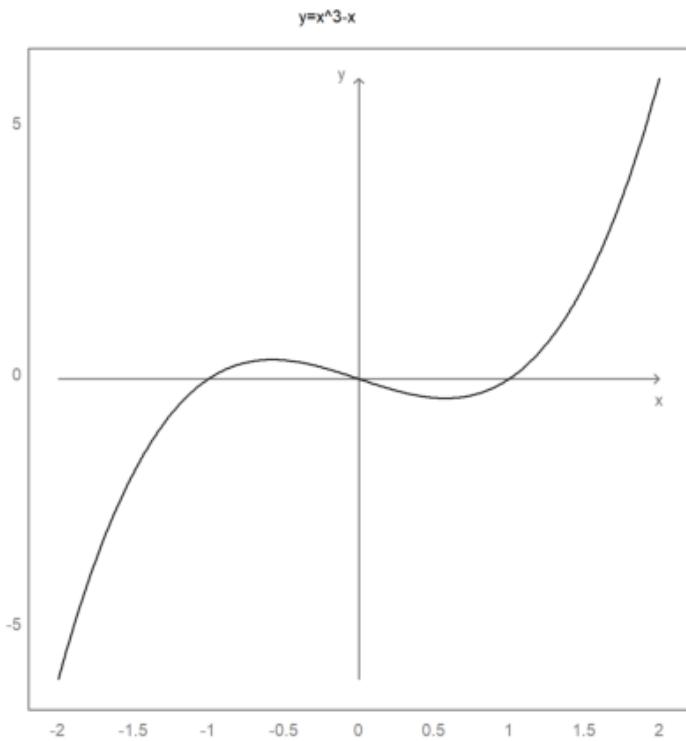
```
>plot2d(&f(x),-1,1,-2,2); // function
>plot2d(&diff(f(x),x),>add,style="--",color=red); // derivative
>labelbox(["f","f'"],["-", "--"],[black,red]): // label box
```



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



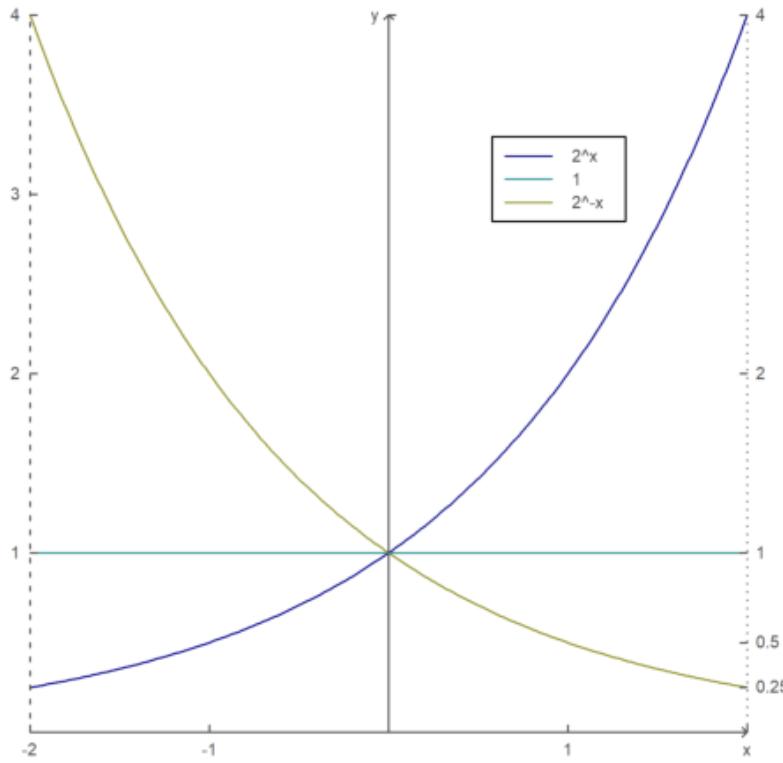
```
>gridstyle("->", color=gray, textcolor=gray, framecolor=gray); ...
> plot2d("x^3-x", grid=1); ...
> setttitle("y=x^3-x", color=black); ...
> label("x", 2, 0, pos="bc", color=gray); ...
> label("y", 0, 6, pos="cl", color=gray); ...
> reset():
```



Untuk kontrol yang lebih banyak lagi, sumbu x dan sumbu y dapat dilakukan secara manual.

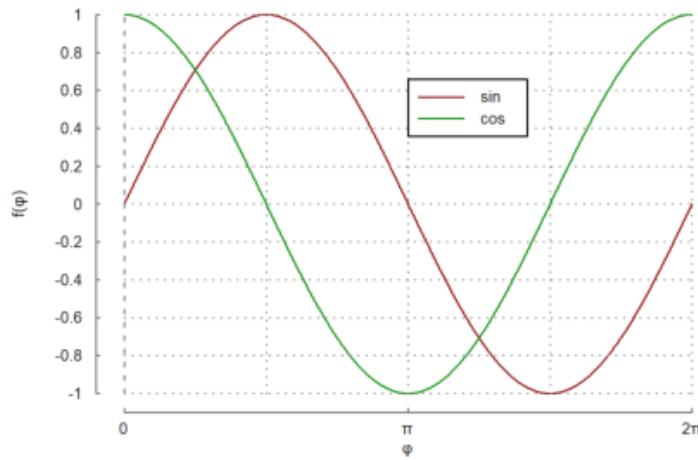
Perintah fullwindow() memperluas jendela plot karena kita tidak lagi membutuhkan tempat untuk label di luar jendela plot. Gunakan shrinkwindow() atau reset() untuk mengatur ulang ke default.

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



Berikut ini adalah contoh lain, di mana string Unicode digunakan dan sumbu di luar area plot.

```
>aspect(1.5);
>plot2d(["sin(x)","cos(x")],0,2pi,color=[red,green],<grid,<frame); ...
>xaxis(-1.1,(0:2)*pi,xt=["0",u"\u03c0;","u"2\u03c0;"],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"\u03c6"); ylabel(u"f(\u03c6)"):
```



Memplot Data 2D

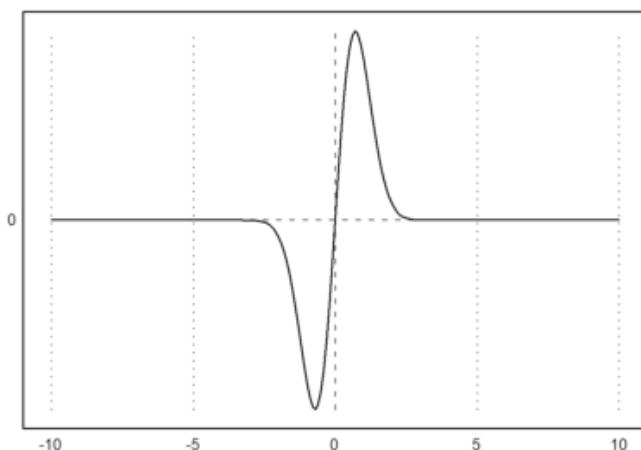
Jika x dan y adalah vektor data, data ini akan digunakan sebagai koordinat x dan y dari sebuah kurva. Dalam hal ini, a , b , c , dan d , atau radius r dapat ditentukan, atau jendela plot akan menyesuaikan secara otomatis dengan data. Atau, `>square` dapat diatur untuk mempertahankan rasio aspek persegi.

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

Tetapi Anda dapat memasukkan data secara langsung.

- Gunakan vektor baris untuk x dan y untuk satu fungsi.
- Matriks untuk x dan y diplot baris demi baris. Berikut adalah contoh dengan satu baris untuk x dan y .

```
>x=-10:0.1:10; y=exp(-x^2)*x; plot2d(x,y);
```



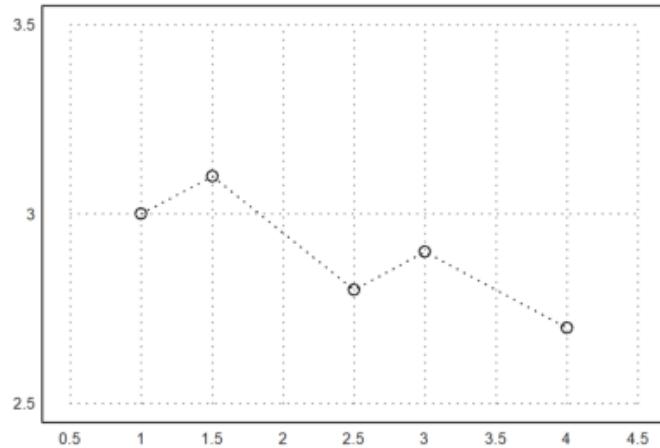
Data juga dapat diplot sebagai titik. Gunakan `poin=true` untuk ini. Plot ini berfungsi seperti poligon, namun hanya menggambar sudut-sudutnya saja.

- `style = "...":` Pilih dari "[]", "<>", "o", ".", "..", "+", "*", "[]", "<>", "o", "..", "", "|".

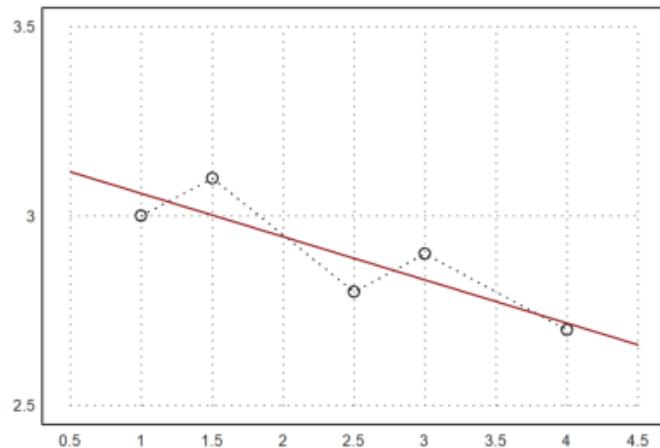
Untuk memplot kumpulan titik, gunakan `>titik`. Jika warna adalah vektor warna, setiap titik mendapatkan warna yang berbeda. Untuk matriks koordinat dan vektor kolom, warna berlaku untuk baris matriks.

Parameter `>addpoints` menambahkan titik ke segmen garis untuk plot data.

```
>xdata=[1,1.5,2.5,3,4]; ydata=[3,3.1,2.8,2.9,2.7]; // data
>plot2d(xdata,ydata,a=0.5,b=4.5,c=2.5,d=3.5,style="."); // lines
>plot2d(xdata,ydata,>points,>add,style="o"); // add points
```



```
>p=polyfit(xdata,ydata,1); // get regression line
>plot2d("polyval(p,x)",>add,color=red); // add plot of line
```



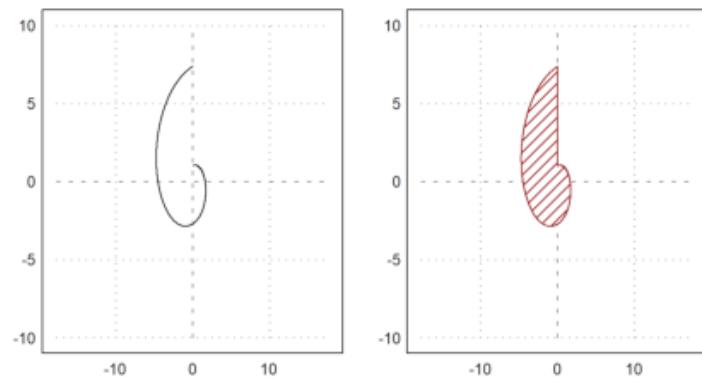
Menggambar Daerah Yang Dibatasi Kurva

Plot data benar-benar berupa poligon. Kita juga dapat memplot kurva atau kurva terisi.

- filled = true mengisi plot.
- style = "...": Pilih dari "", "/", "\", "\/".
- fillcolor: Lihat di atas untuk warna yang tersedia.

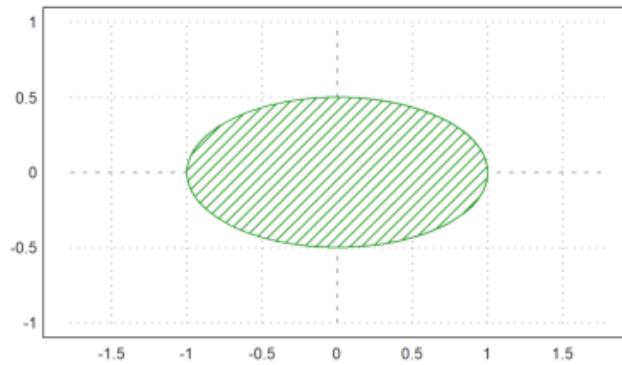
Warna isian ditentukan oleh argumen "fillcolor", dan pada opsional <outline mencegah menggambar garis batas untuk semua gaya kecuali yang default.

```
>t=linspace(0,2pi,1000); // parameter for curve
>x=sin(t)*exp(t/pi); y=cos(t)*exp(t/pi); // x(t) and y(t)
>figure(1,2); aspect(16/9)
>figure(1); plot2d(x,y,r=10); // plot curve
>figure(2); plot2d(x,y,r=10,>filled,style="/",fillcolor=red); // fill curve
>figure(0):
```

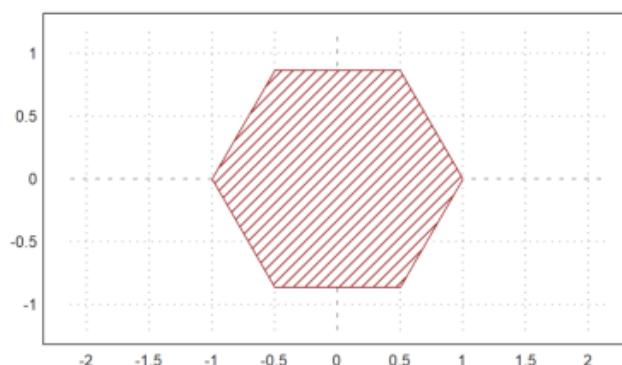


Pada contoh berikut ini, kami memplot elips yang terisi dan dua segi enam yang terisi menggunakan kurva tertutup dengan 6 titik dengan gaya isian yang berbeda.

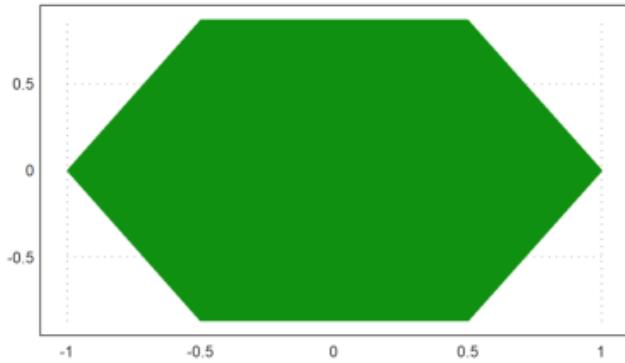
```
>x=linspace(0,2pi,1000); plot2d(sin(x),cos(x)*0.5,r=1,>filled,style="/"):
```



```
>t=linspace(0,2pi,6); ...
>plot2d(cos(t),sin(t),>filled,style="/",fillcolor=red,r=1.2):
```

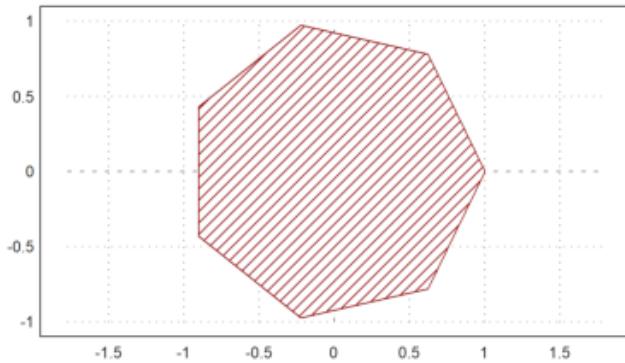


```
>t=linspace(0,2pi,6); plot2d(cos(t),sin(t),>filled,style="#"):
```



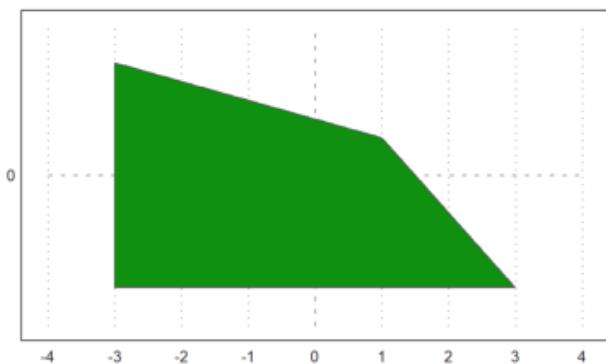
Contoh lainnya adalah septagon, yang kita buat dengan 7 titik pada lingkaran satuan.

```
>t=linspace(0,2pi,7); ...
> plot2d(cos(t),sin(t),r=1,>filled,style="/",fillcolor=red):
```



Berikut ini adalah himpunan nilai maksimal dari empat kondisi linier yang kurang dari atau sama dengan 3. Ini adalah $A[k].v \leq 3$ untuk semua barisan A . Untuk mendapatkan sudut-sudut yang bagus, kita menggunakan n yang relatif besar.

```
>A=[2,1;1,2;-1,0;0,-1];
>function f(x,y) := max([x,y].A');
>plot2d("f",r=4,level=[0;3],color=green,n=111):
```

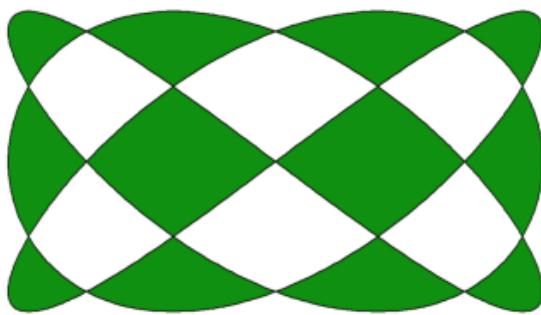


Poin utama dari bahasa matriks adalah bahwa bahasa ini memungkinkan untuk menghasilkan tabel fungsi dengan mudah.

```
>t=linspace(0,2pi,1000); x=cos(3*t); y=sin(4*t);
```

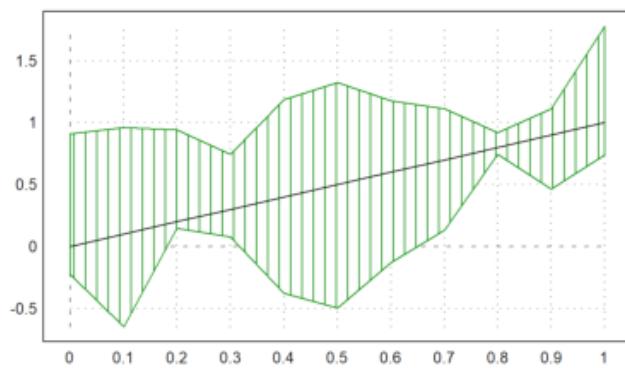
Kita sekarang memiliki vektor nilai x dan y. plot2d() dapat memplot nilai-nilai ini sebagai sebuah kurva yang menghubungkan titik-titik. Plot dapat diisi. Dalam kasus ini, hal ini memberikan hasil yang bagus karena aturan penggulungan, yang digunakan untuk isi.

```
>plot2d(x,y,<grid,<frame,>filled):
```



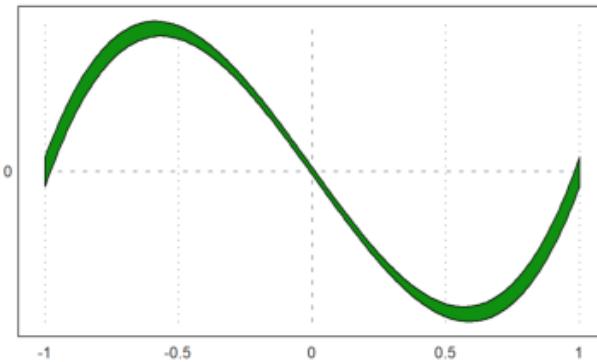
Vektor interval diplot terhadap nilai x sebagai wilayah yang terisi antara nilai bawah dan atas interval. Hal ini dapat berguna untuk memplot kesalahan perhitungan. Tetapi juga dapat digunakan untuk memplot kesalahan statistik.

```
>t=0:0.1:1; ...
> plot2d(t,interval(t-random(size(t)),t+random(size(t))),style="|"); ...
> plot2d(t,t,add=true):
```



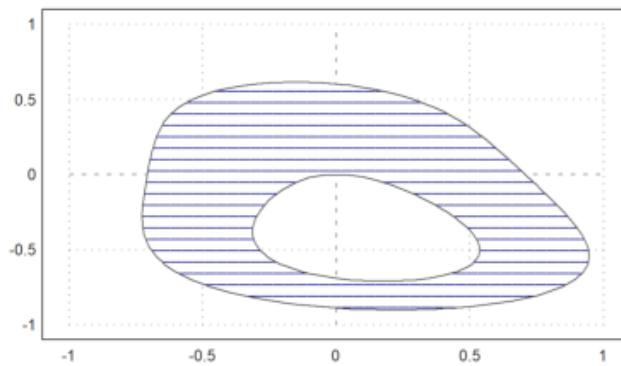
Jika x adalah vektor yang diurutkan, dan y adalah vektor interval, maka plot2d akan memplot rentang interval yang terisi di bidang, gaya isian sama dengan gaya poligon.

```
>t=-1:0.01:1; x=~t-0.01,t+0.01~; y=x^3-x;
>plot2d(t,y):
```



Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks 2xn. Baris pertama adalah batas bawah dan baris kedua berisi batas atas.

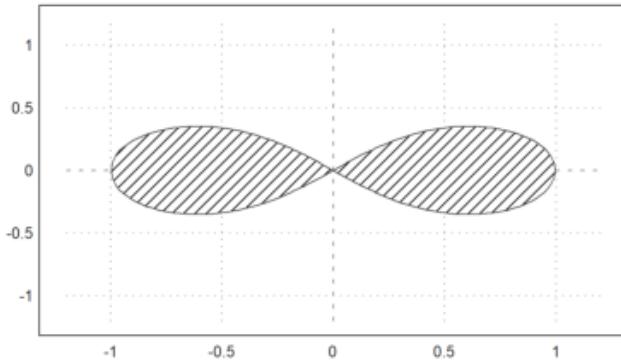
```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <= 1
```



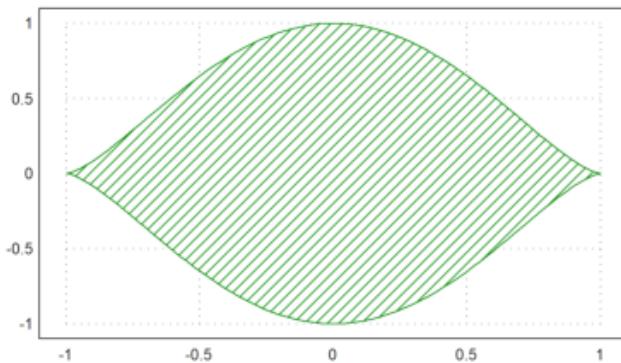
Kita juga dapat mengisi rentang nilai seperti

$$-1 \leq (x^2 + y^2)^2 - x^2 + y^2 \leq 0.$$

```
>plot2d("(x^2+y^2)^2-x^2+y^2",r=1.2,level=[-1;0],style="/"):
```



```
>plot2d("cos(x)", "sin(x)^3", xmin=0, xmax=2pi, >filled, style="/"):
```



Grafik Fungsi Parametrik

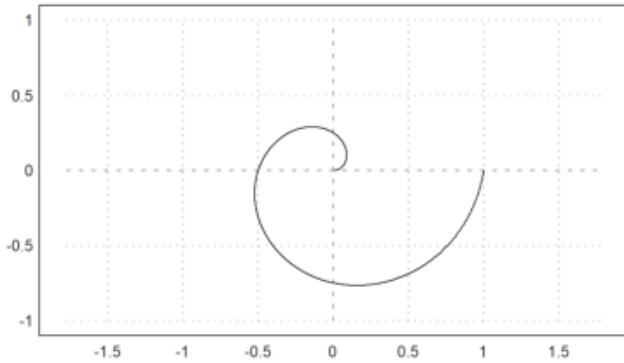
Nilai x tidak perlu diurutkan. (x,y) hanya menggambarkan sebuah kurva. Jika x diurutkan, kurva tersebut adalah grafik dari sebuah fungsi.

Dalam contoh berikut, kami memplot spiral

$$\gamma(t) = t \cdot (\cos(2\pi t), \sin(2\pi t))$$

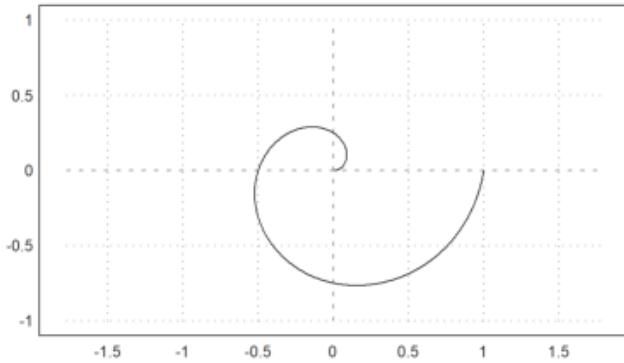
Kita perlu menggunakan sangat banyak titik untuk tampilan yang halus atau fungsi adaptive() untuk mengevaluasi ekspresi (lihat fungsi adaptive() untuk lebih jelasnya).

```
>t=linspace(0,1,1000); ...
>plot2d(t*cos(2*pi*t),t*sin(2*pi*t),r=1):
```

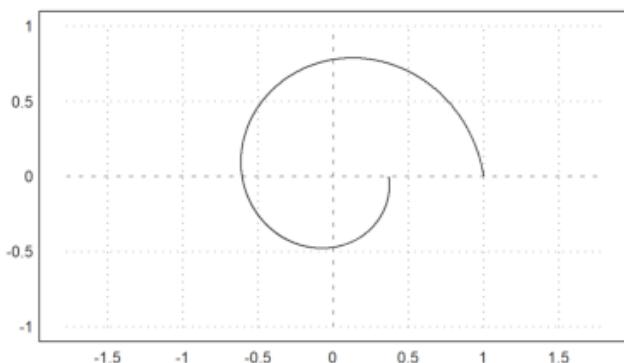


Sebagai alternatif, Anda dapat menggunakan dua ekspresi untuk kurva. Berikut ini memplot kurva yang sama seperti di atas.

```
>plot2d("x*cos(2*pi*x)", "x*sin(2*pi*x)", xmin=0, xmax=1, r=1):
```



```
>t=linspace(0,1,1000); r=exp(-t); x=r*cos(2pi*t); y=r*sin(2pi*t);
>plot2d(x,y,r=1):
```



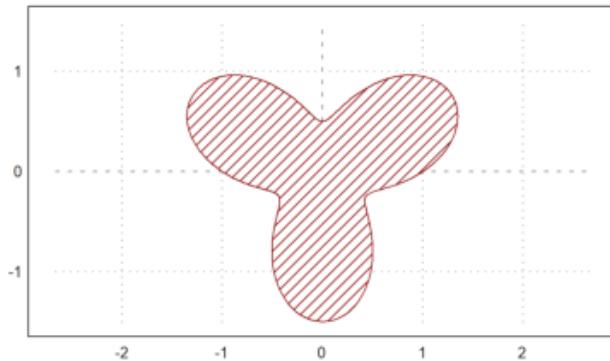
Dalam contoh berikut, kami memplot kurva

$$\gamma(t) = (r(t) \cos(t), r(t) \sin(t))$$

dengan

$$r(t) = 1 + \frac{\sin(3t)}{2}.$$

```
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t); ...
>plot2d(x,y,>filled,fillcolor=red,style="/"',r=1.5):
```



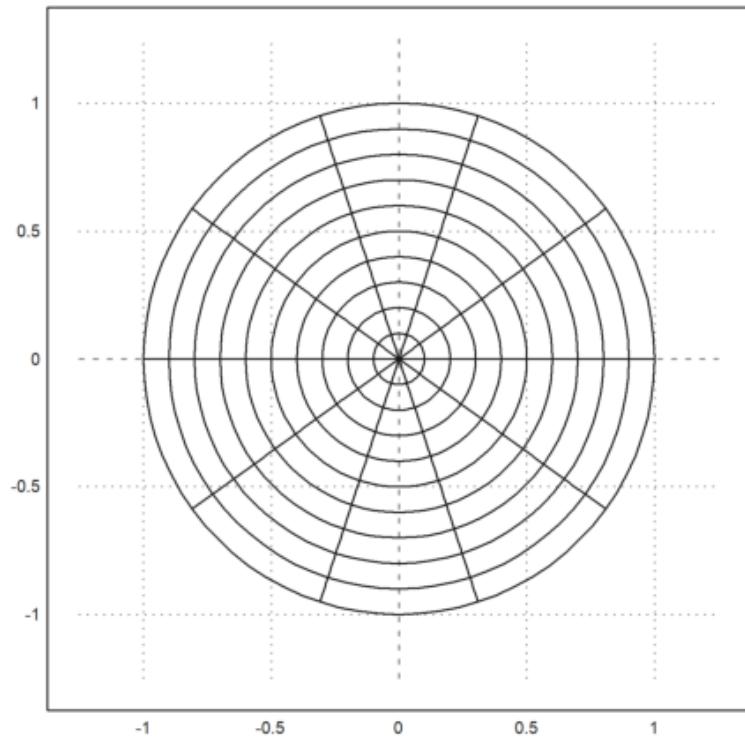
Menggambar Grafik Bilangan Kompleks

Larik bilangan kompleks juga dapat diplot. Kemudian titik-titik kisi akan dihubungkan. Jika sejumlah garis kisi ditentukan (atau vektor 1×2 garis kisi) pada argumen cgrid, hanya garis-garis kisi tersebut yang akan terlihat.

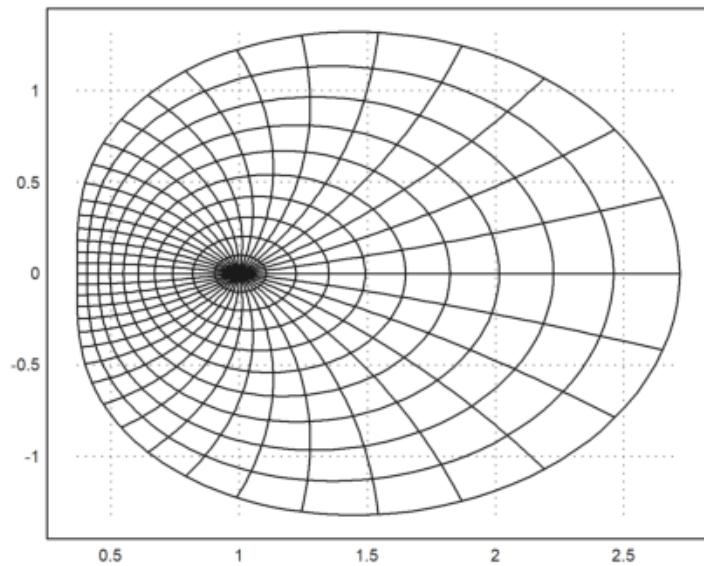
Matriks bilangan kompleks akan secara otomatis diplot sebagai kisi-kisi pada bidang kompleks.

Pada contoh berikut, kami memplot gambar lingkaran satuan di bawah fungsi eksponensial. Parameter cgrid menyembunyikan beberapa kurva kisi-kisi.

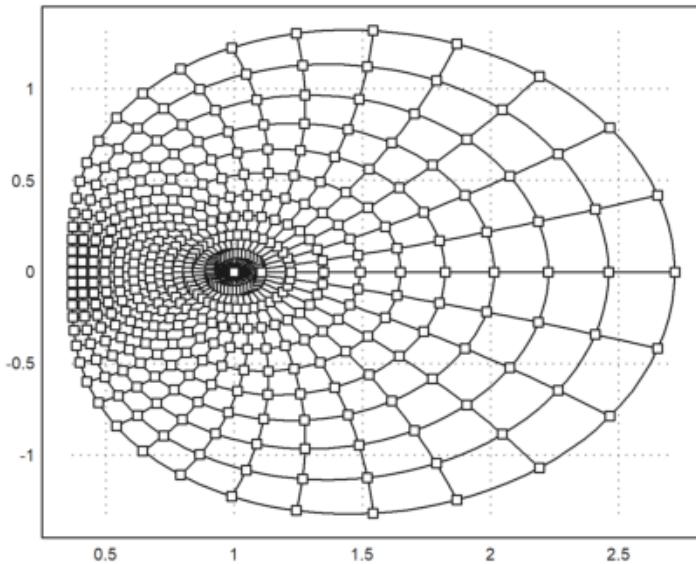
```
>aspect(); r=linspace(0,1,50); a=linspace(0,2pi,80)'; z=r*exp(I*a);...
>plot2d(z,a=-1.25,b=1.25,c=-1.25,d=1.25,cgrid=10):
```



```
>aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)'; z=r*exp(I*a);
>plot2d(exp(z),cgrid=[40,10]):
```



```
>r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=r*exp(I*a);
>plot2d(exp(z),>points,>add):
```

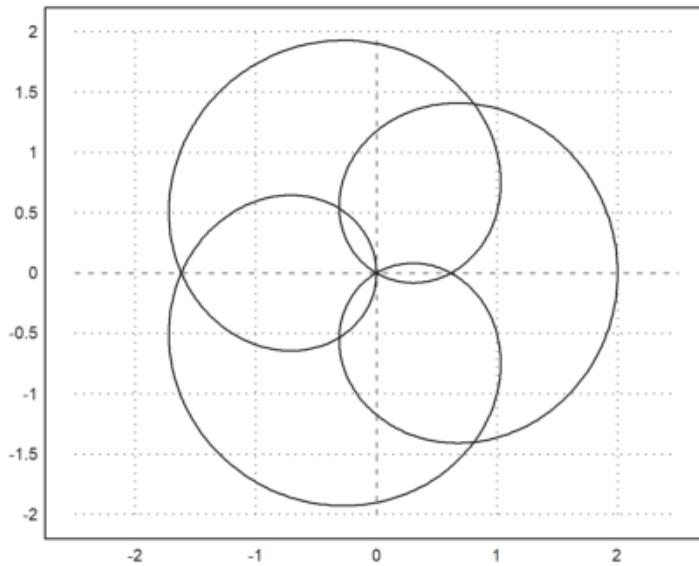


Vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian riil dan bagian imajiner.

Dalam contoh, kami memplot lingkaran satuan dengan

$$\gamma(t) = e^{it}$$

```
>t=linspace(0,2pi,1000); ...
>plot2d(exp(I*t)+exp(4*I*t),r=2);
```



Plot Statistik

Terdapat banyak fungsi yang dikhkususkan pada plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

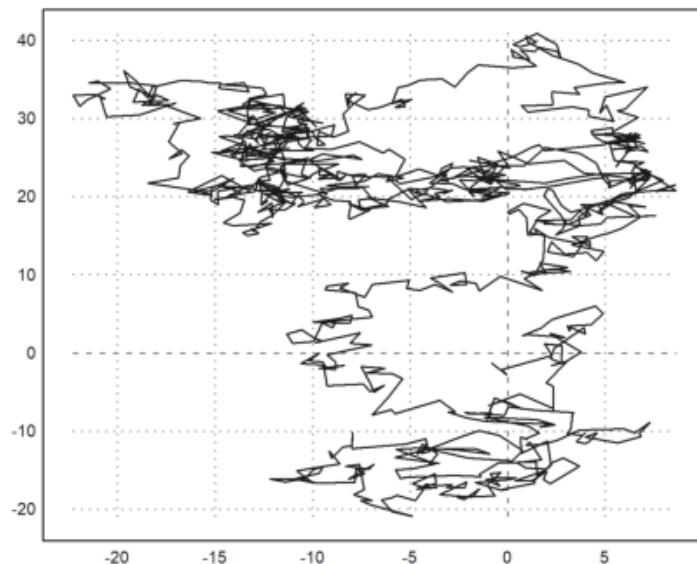
Jumlah kumulatif dari nilai berdistribusi normal 0-1 menghasilkan jalan acak.

```
>plot2d(cumsum(randnormal(1,1000))):
```

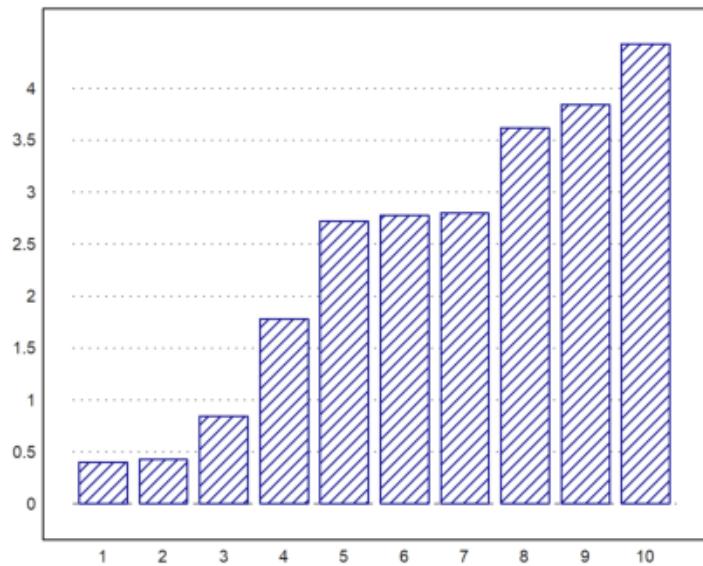


Dengan menggunakan dua baris, ini menunjukkan jalan kaki dalam dua dimensi.

```
>X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]):
```

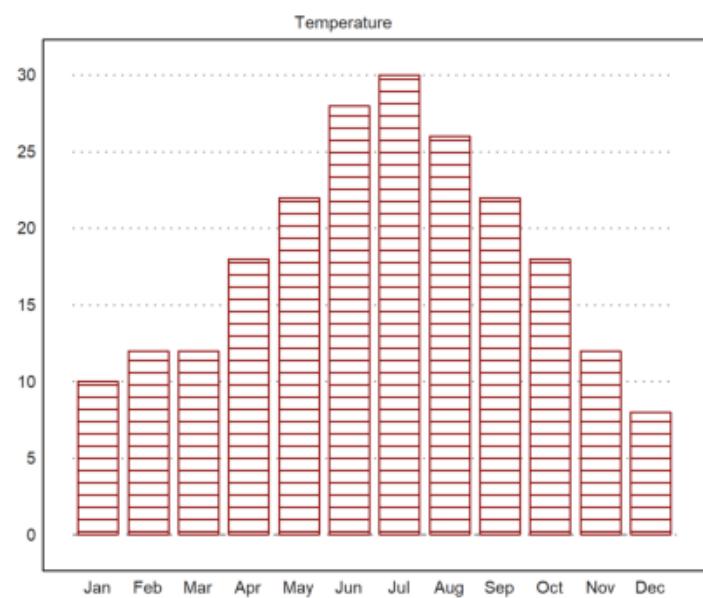


```
>columnsplot(cumsum(random(10)),style="/",color=blue):
```

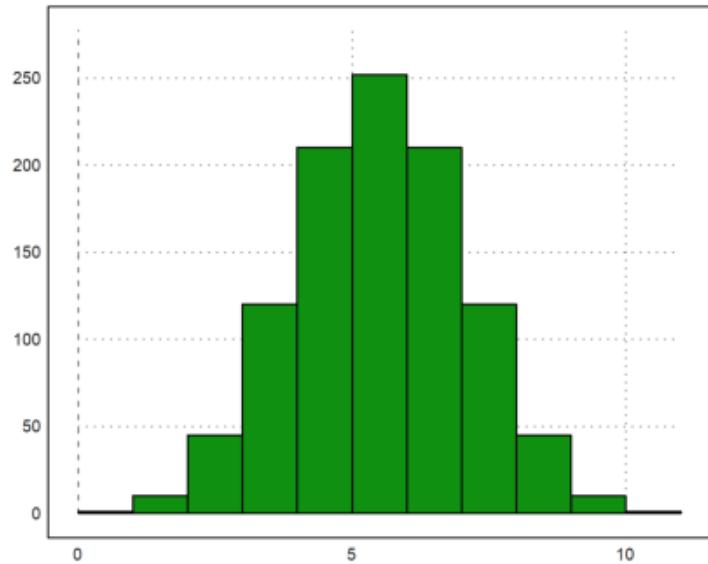


Ini juga dapat menampilkan string sebagai label.

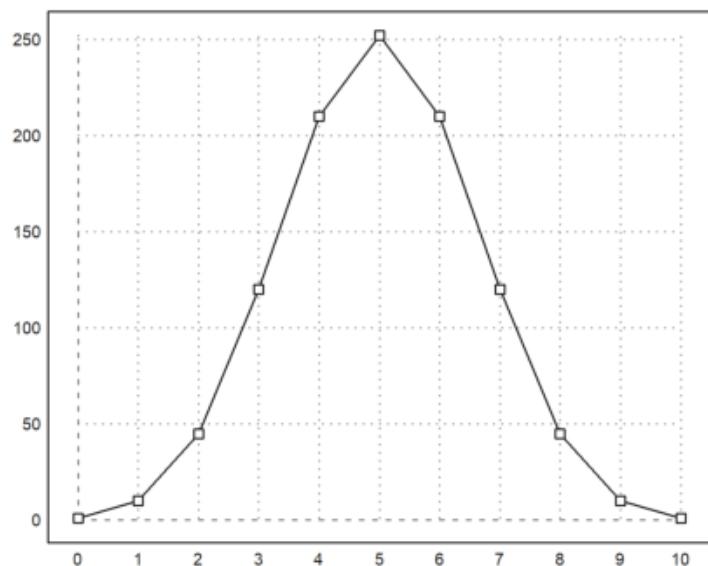
```
>months=["Jan", "Feb", "Mar", "Apr", "May", "Jun", ...
> "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"];
>values=[10,12,12,18,22,28,30,26,22,18,12,8];
>columnsplot(values,lab=months,color=red,style="-");
>title("Temperature"):
```



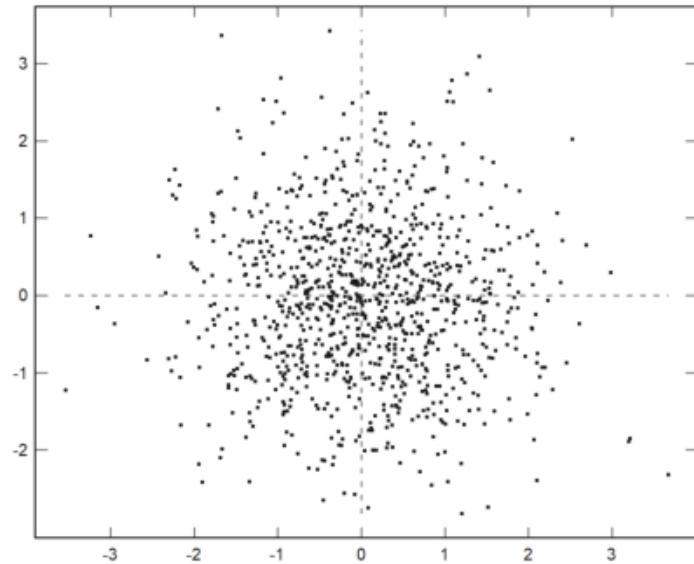
```
>k=0:10;
>plot2d(k,bin(10,k),>bar):
```



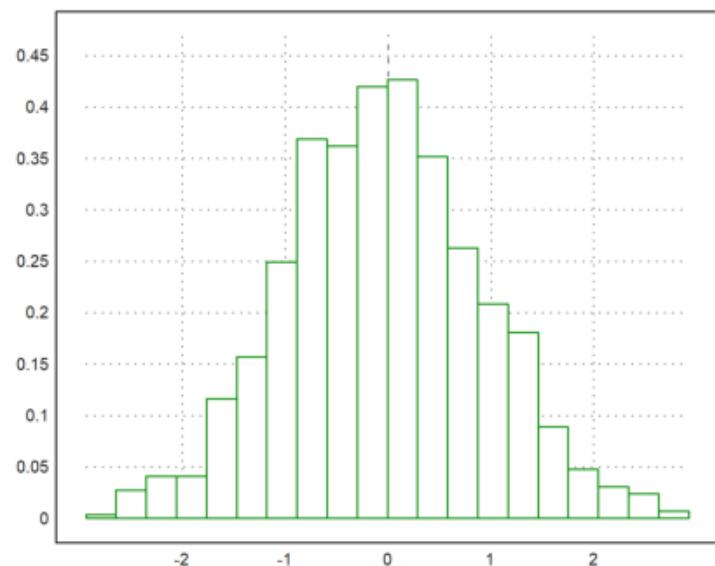
```
>plot2d(k,bin(10,k)); plot2d(k,bin(10,k),>points,>add):
```



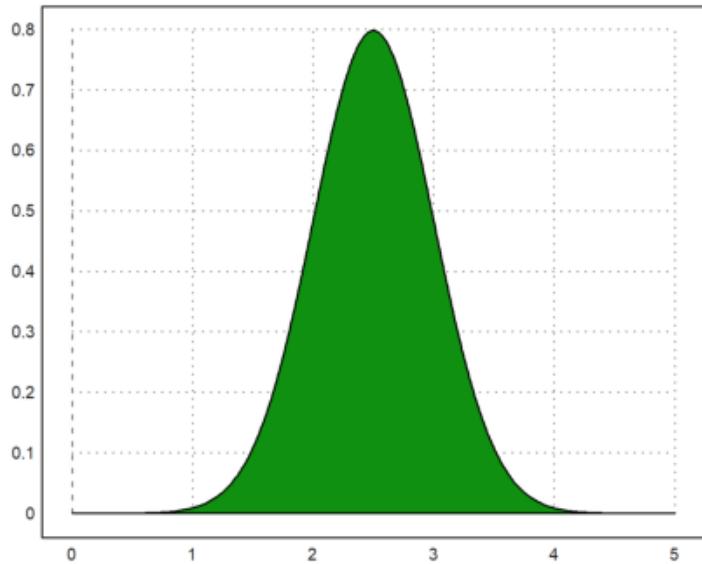
```
>plot2d(normal(1000),normal(1000),>points,grid=6,style="."):
```



```
>plot2d(normal(1,1000),>distribution,style="O"):
```

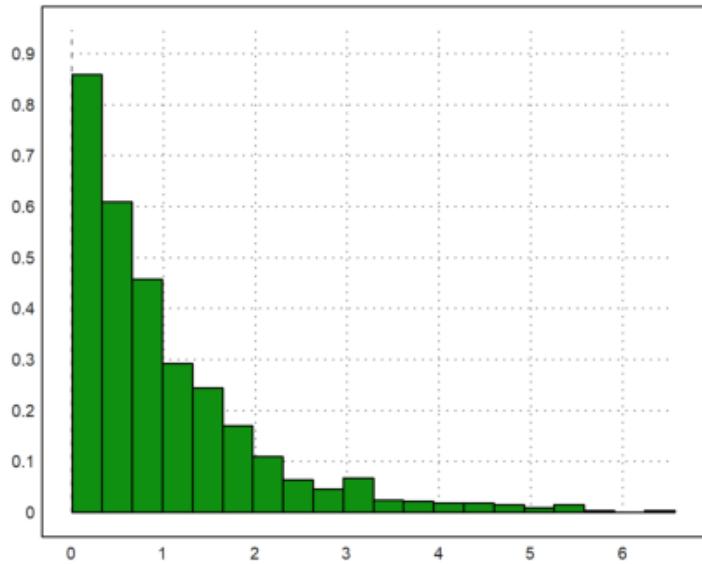


```
>plot2d("qnormal",0,5;2.5,0.5,>filled):
```



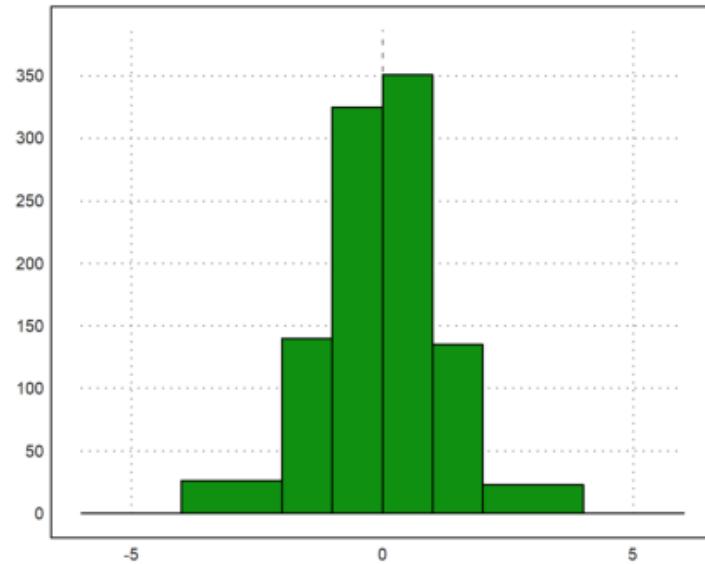
Untuk memplot distribusi statistik eksperimental, Anda dapat menggunakan distribution=n dengan plot2d.

```
>w=randexponential(1,1000); // exponential distribution
>plot2d(w,>distribution); // or distribution=n with n intervals
```



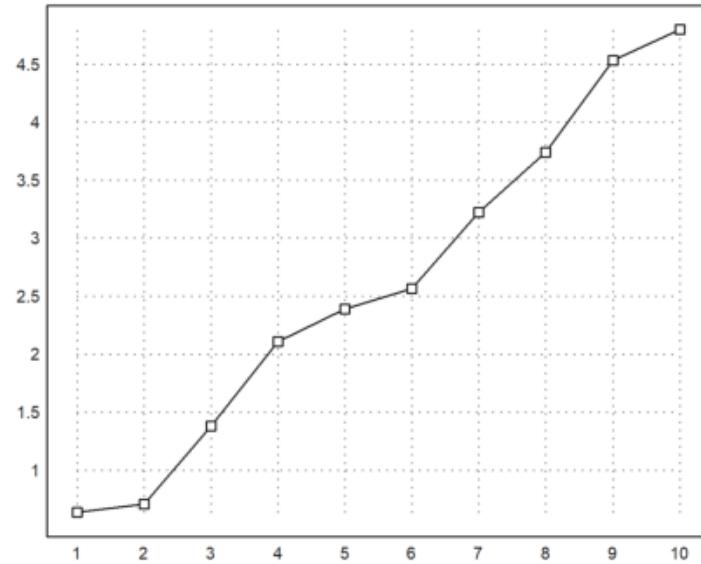
Atau Anda dapat menghitung distribusi dari data dan memplot hasilnya dengan >bar di plot3d, atau dengan plot kolom.

```
>w=normal(1000); // 0-1-normal distribution
>{x,y}=histo(w,10,v=[-6,-4,-2,-1,0,1,2,4,6]); // interval bounds v
>plot2d(x,y,>bar):
```

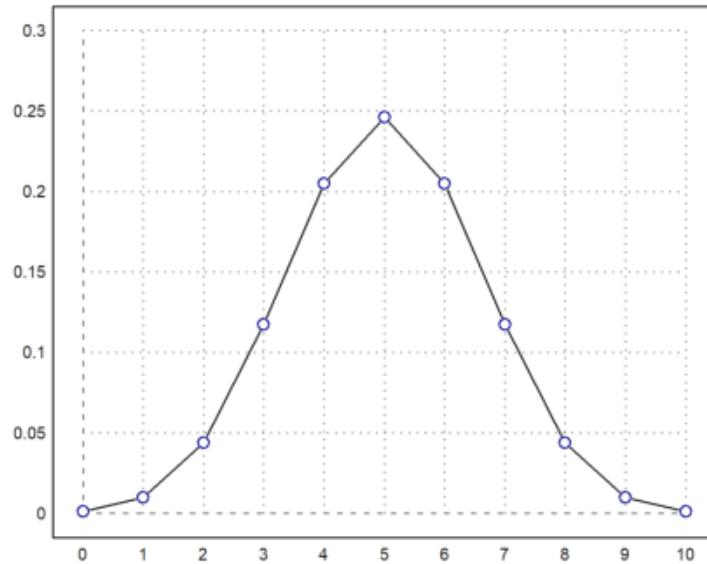


Fungsi statplot() menetapkan gaya dengan string sederhana.

```
>statplot(1:10,cumsum(random(10)), "b"):
```



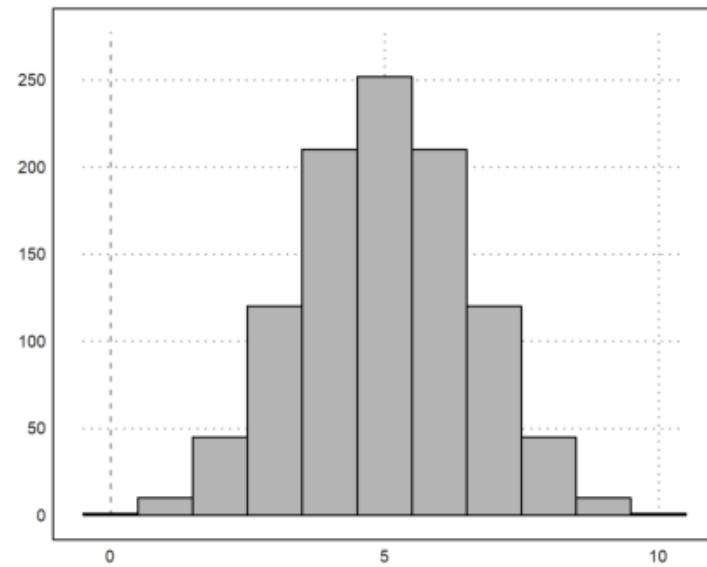
```
>n=10; i=0:n; ...
>plot2d(i,bin(n,i)/2^n,a=0,b=10,c=0,d=0.3); ...
>plot2d(i,bin(n,i)/2^n,points=true,style="ow",add=true,color=blue):
```



Selain itu, data dapat diplot sebagai batang. Dalam hal ini, x harus diurutkan dan satu elemen lebih panjang dari y. Batang akan memanjang dari $x[i]$ ke $x[i+1]$ dengan nilai $y[i]$. Jika x memiliki ukuran yang sama dengan y, maka x akan diperpanjang satu elemen dengan jarak terakhir.

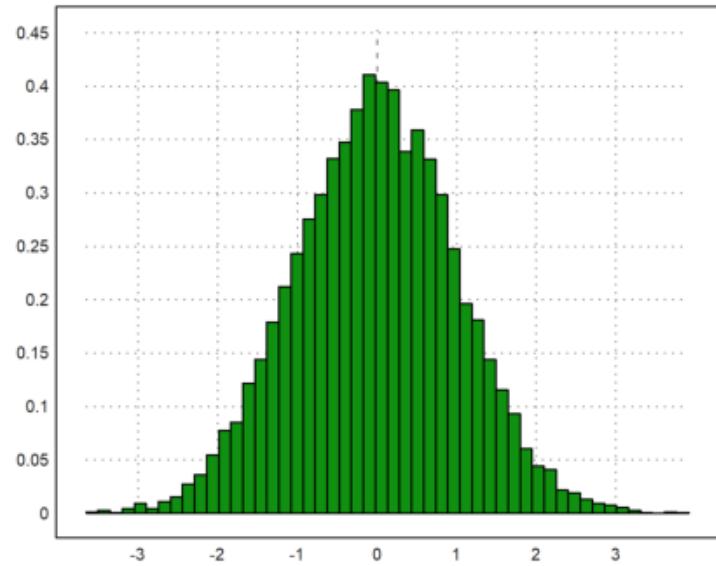
Gaya isian dapat digunakan seperti di atas.

```
>n=10; k=bin(n,0:n); ...
>plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray):
```

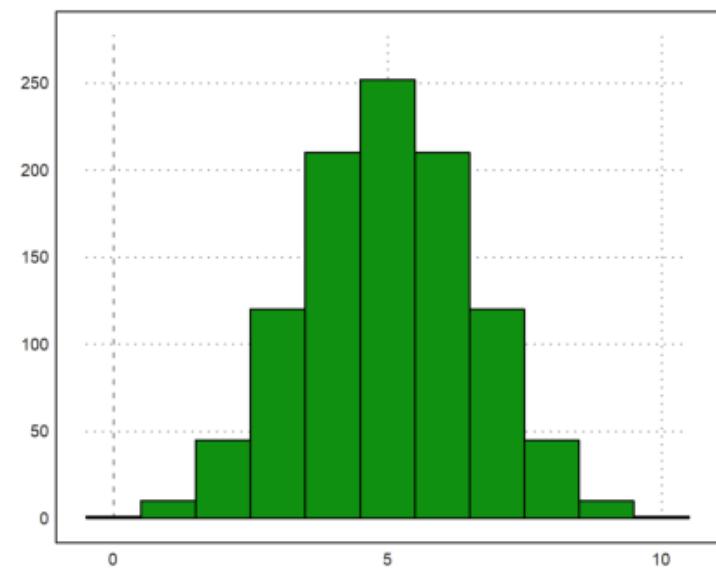


Data untuk plot batang (batang = 1) dan histogram (histogram = 1) dapat diberikan secara eksplisit dalam xv dan yv, atau dapat dihitung dari distribusi empiris dalam xv dengan >distribusi (atau distribusi = n). Histogram dari nilai xv akan dihitung secara otomatis dengan >histogram. Jika >even ditentukan, nilai xv akan dihitung dalam interval bilangan bulat.

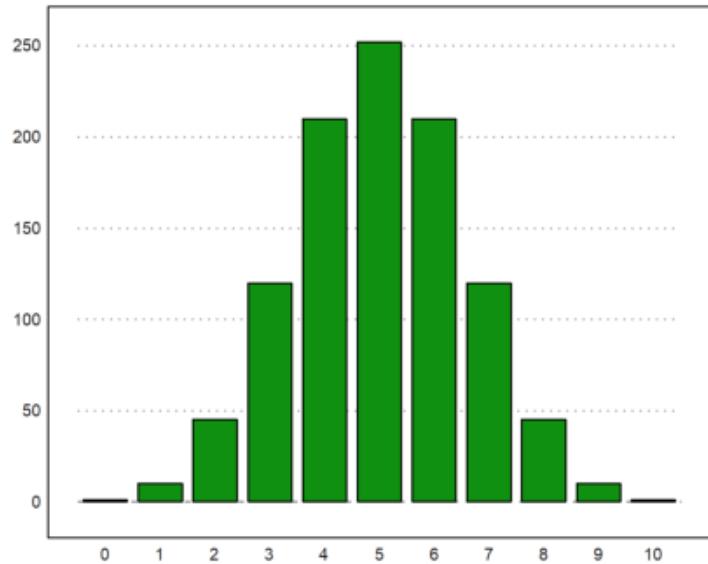
```
>plot2d(normal(10000),distribution=50):
```



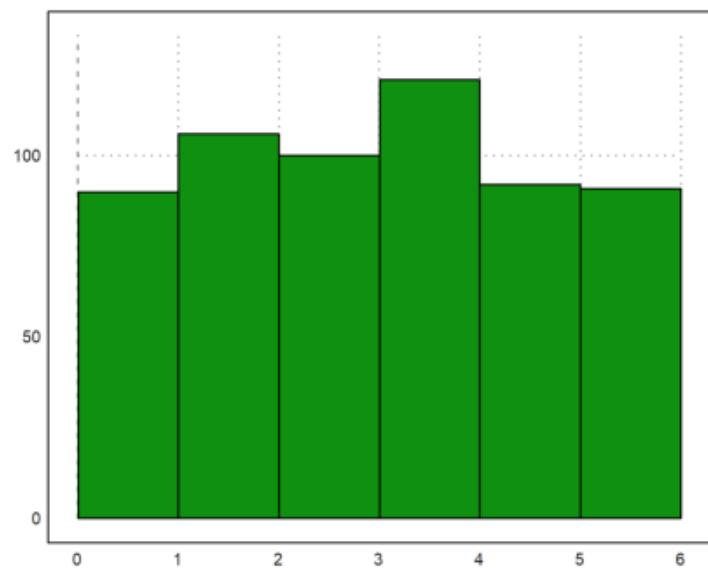
```
>k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m,>bar):
```



```
>columnsplot(m,k):
```

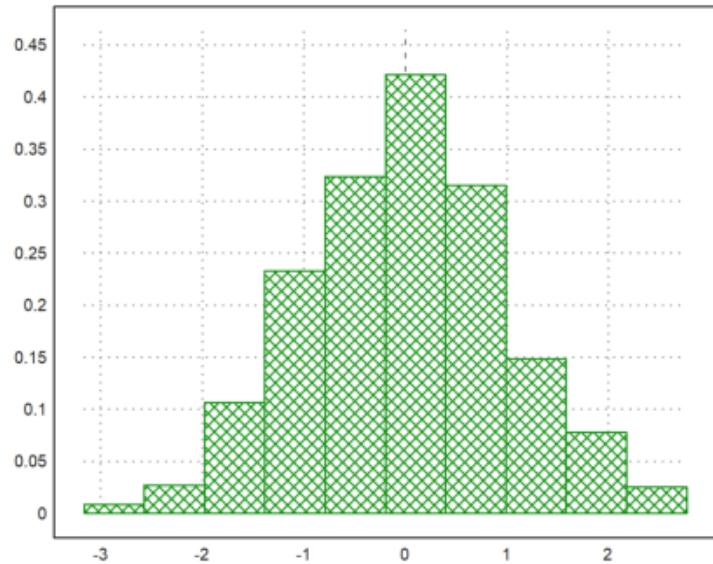


```
>plot2d(random(600)*6,histogram=6):
```



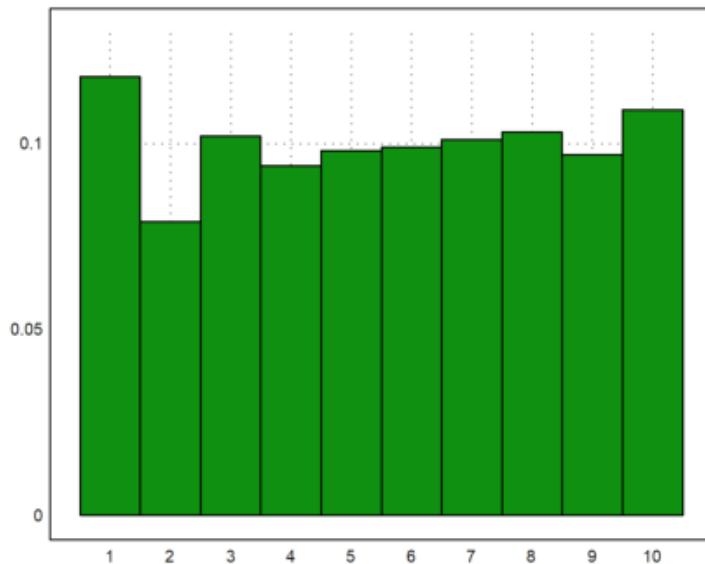
Untuk distribusi, ada parameter `distribution=n`, yang menghitung nilai secara otomatis dan mencetak distribusi relatif dengan n sub-interval.

```
>plot2d(normal(1,1000),distribution=10,style="\\"/"):
```



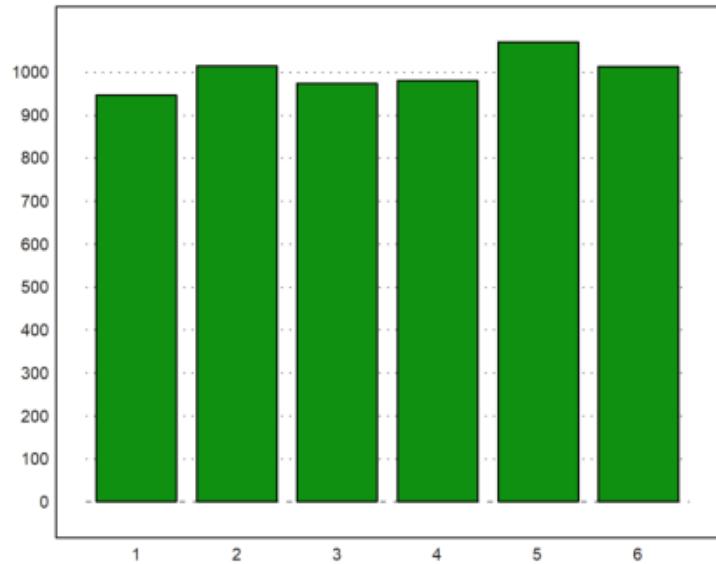
Dengan parameter even=true, ini akan menggunakan interval bilangan bulat.

```
>plot2d(intrandom(1,1000,10),distribution=10,even=true) :
```

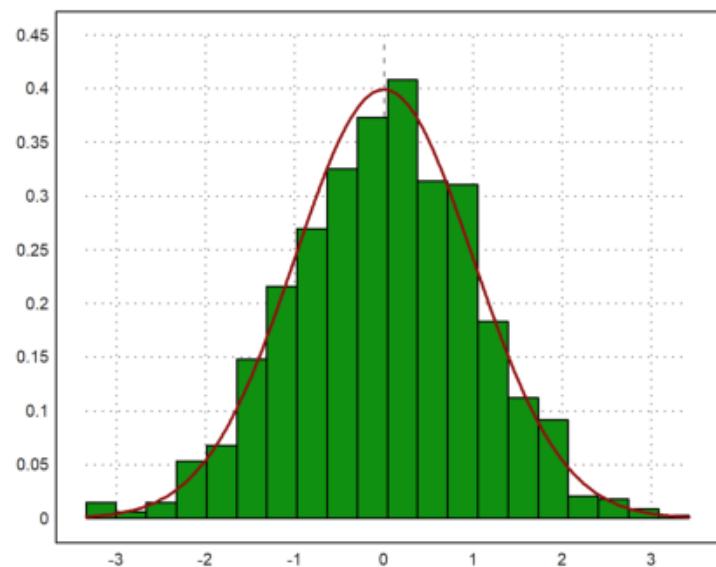


Perhatikan bahwa ada banyak plot statistik yang mungkin berguna. Lihatlah tutorial tentang statistik.

```
>columnspplot(getmultiplicities(1:6,intrandom(1,6000,6))) :
```

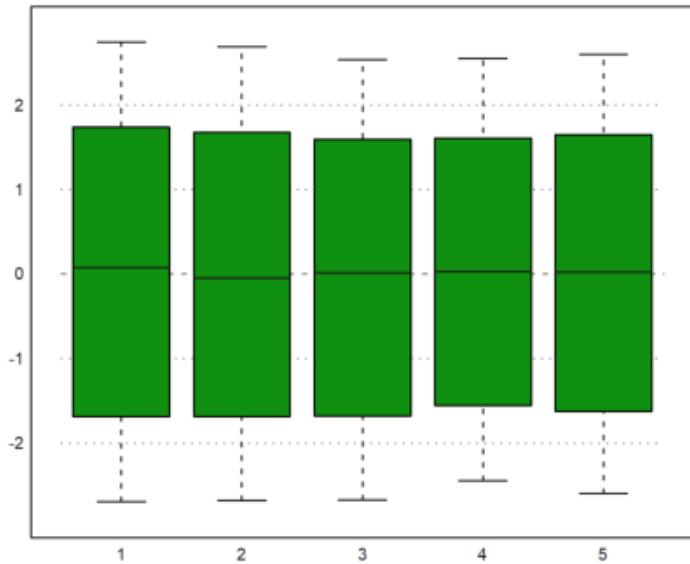


```
>plot2d(normal(1,1000),>distribution); ...
> plot2d("qnormal(x)",color=red,thickness=2,>add):
```



Ada juga banyak plot khusus untuk statistik. Boxplot menunjukkan kuartil dari distribusi ini dan banyak pencilan. Menurut definisi, pencilan dalam boxplot adalah data yang melebihi 1,5 kali kisaran 50% tengah plot.

```
>M=normal(5,1000); boxplot(quartiles(M));
```



Fungsi Implisit

Plot implisit menunjukkan garis level yang menyelesaikan $f(x,y)=\text{level}$, di mana "level" dapat berupa nilai tunggal atau vektor nilai. Jika level = "auto", akan ada nc garis level, yang akan menyebar di antara nilai minimum dan maksimum fungsi secara merata. Warna yang lebih gelap atau lebih terang dapat ditambahkan dengan >hue untuk mengindikasikan nilai fungsi. Untuk fungsi implisit, xv haruslah sebuah fungsi atau ekspresi dari parameter x dan y, atau, sebagai alternatif, xv dapat berupa sebuah matriks nilai.

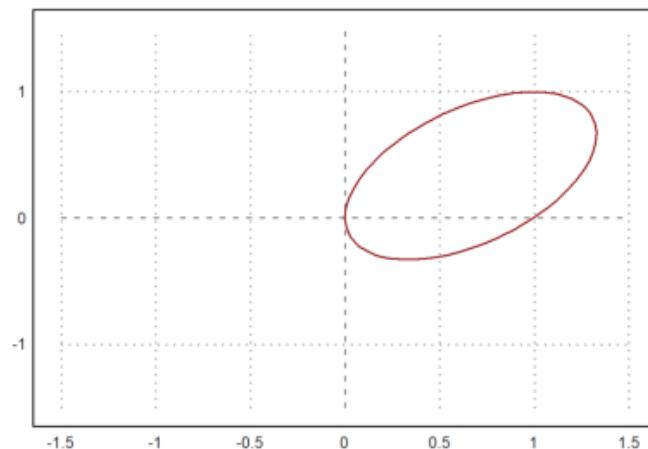
Euler dapat menandai garis level

$$f(x, y) = c$$

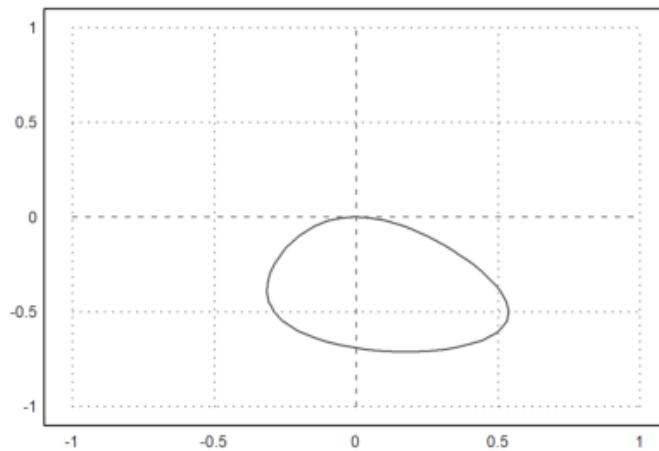
dari fungsi apa pun.

Untuk menggambar himpunan $f(x,y)=c$ untuk satu atau lebih konstanta c , Anda dapat menggunakan `plot2d()` dengan plot implisitnya pada bidang. Parameter untuk c adalah `level = c`, di mana c dapat berupa vektor garis level. Sebagai tambahan, sebuah skema warna dapat digambar pada latar belakang untuk mengindikasikan nilai fungsi untuk setiap titik pada plot. Parameter "n" menentukan kehalusan plot.

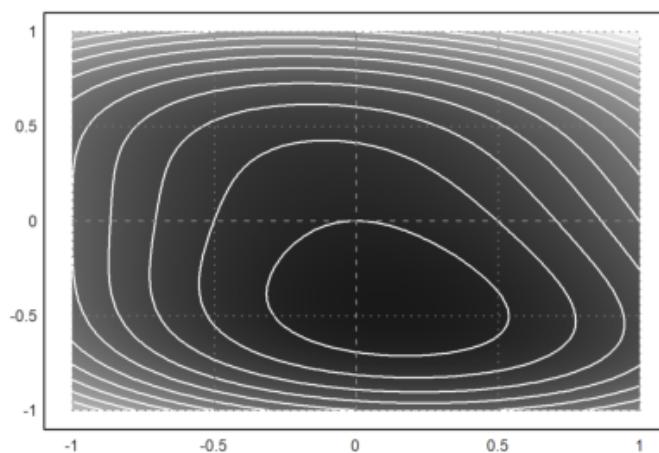
```
>aspect(1.5);
>plot2d("x^2+y^2-x*y-x", r=1.5, level=0, contourcolor=red);
```



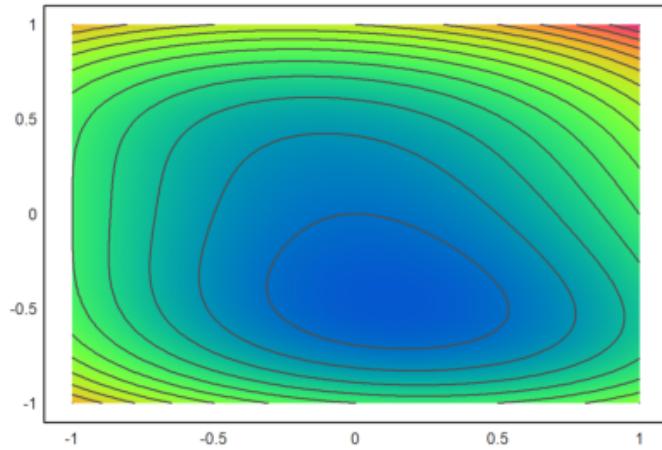
```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)  
>plot2d(expr,level=0); // Solutions of f(x,y)=0
```



```
>plot2d(expr,level=0:0.5:20,>hue,contourcolor=white,n=200); // nice
```

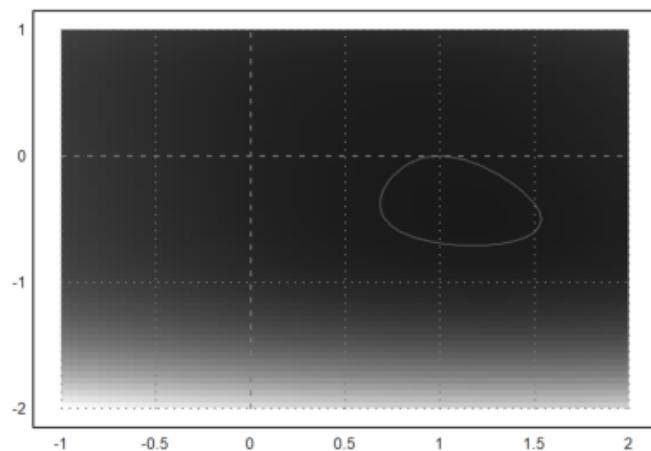


```
>plot2d(expr,level=0:0.5:20,>hue,>spectral,n=200,grid=4); // nicer
```

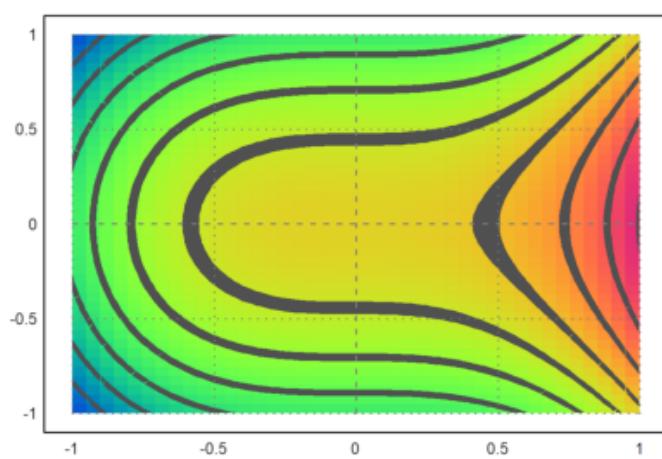


Hal ini juga berlaku untuk plot data. Tetapi Anda harus menentukan rentang untuk label sumbu.

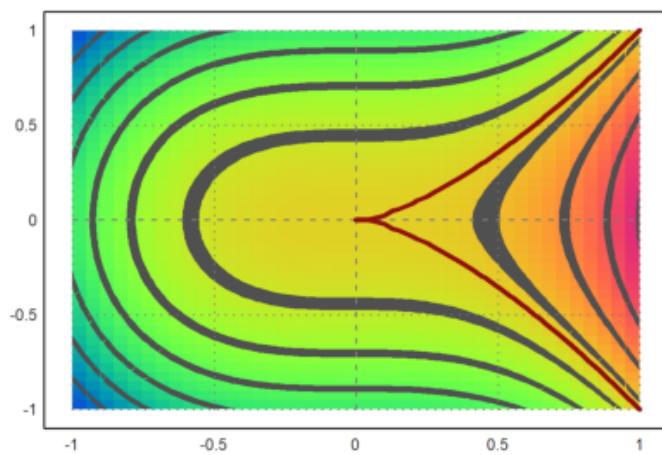
```
>x=-2:0.05:1; y=x'; z=expr(x,y);
>plot2d(z,level=0,a=-1,b=2,c=-2,d=1,>hue):
```



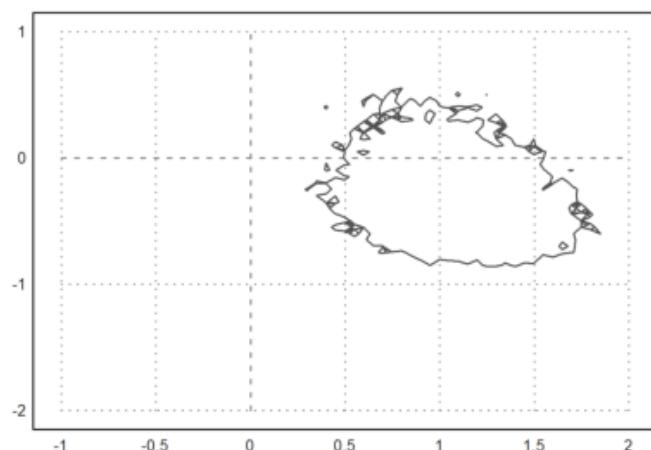
```
>plot2d("x^3-y^2",>contour,>hue,>spectral):
```



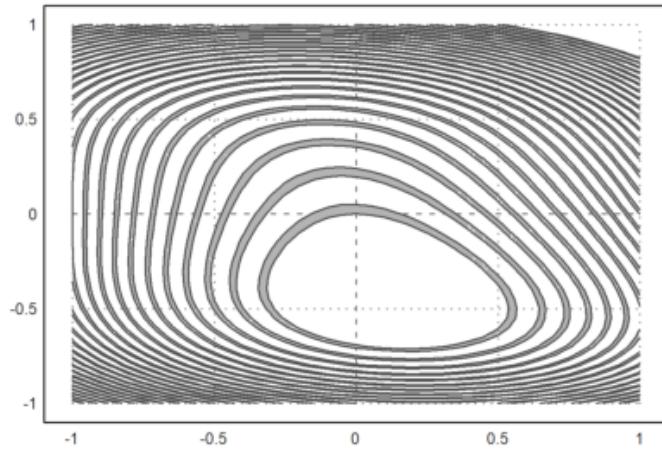
```
>plot2d("x^3-y^2",level=0,contourwidth=3,>add,contourcolor=red):
```



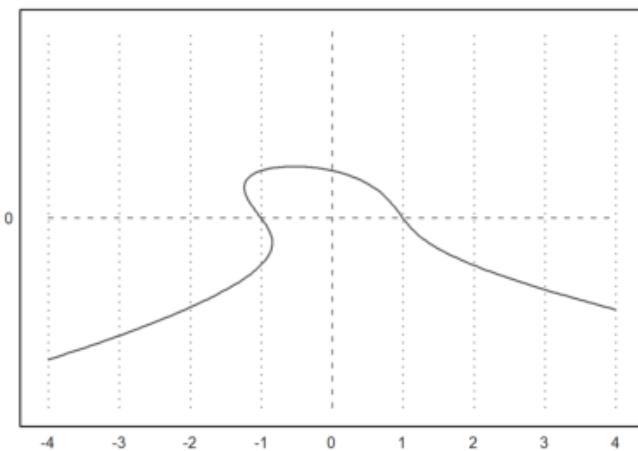
```
>z=z+normal(size(z))*0.2;  
>plot2d(z,level=0.5,a=-1,b=2,c=-2,d=1):
```



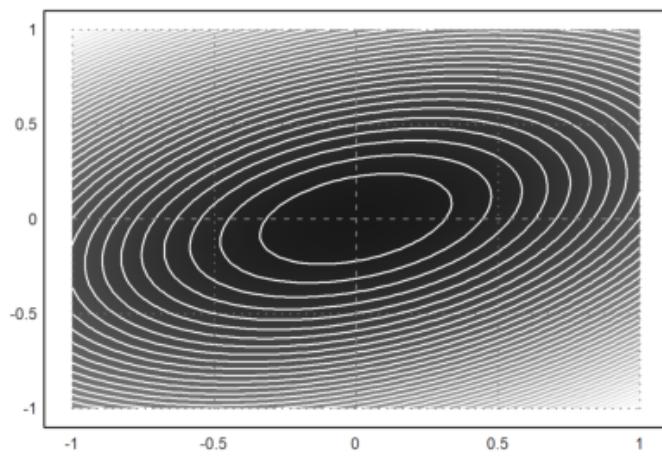
```
>plot2d(expr,level=[0:0.2:5;0.05:0.2:5.05],color=lightgray):
```



```
>plot2d("x^2+y^3+x*y", level=1, r=4, n=100) :
```



```
>plot2d("x^2+2*y^2-x*y", level=0:0.1:10, n=100, contourcolor=white, >hue) :
```



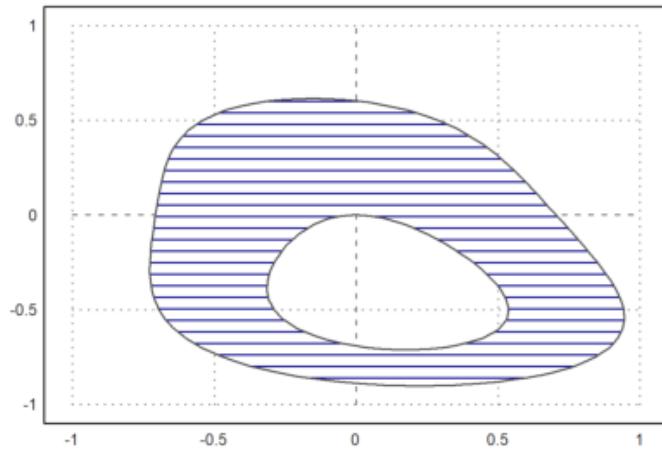
Dimungkinkan juga untuk mengisi set

$$a \leq f(x, y) \leq b$$

dengan rentang level.

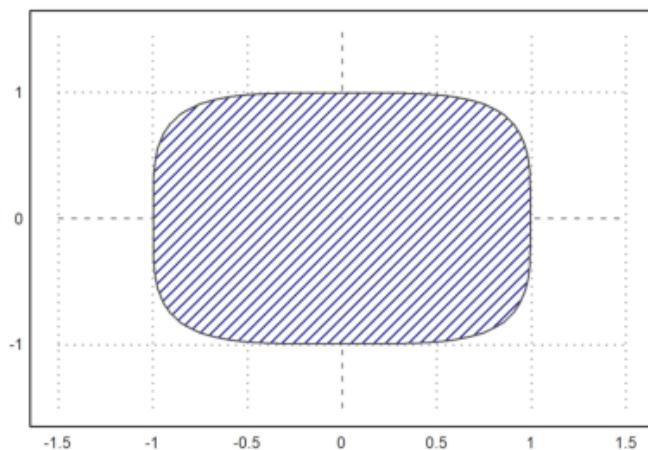
Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks 2xn. Baris pertama adalah batas bawah dan baris kedua berisi batas atas.

```
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <= 1
```

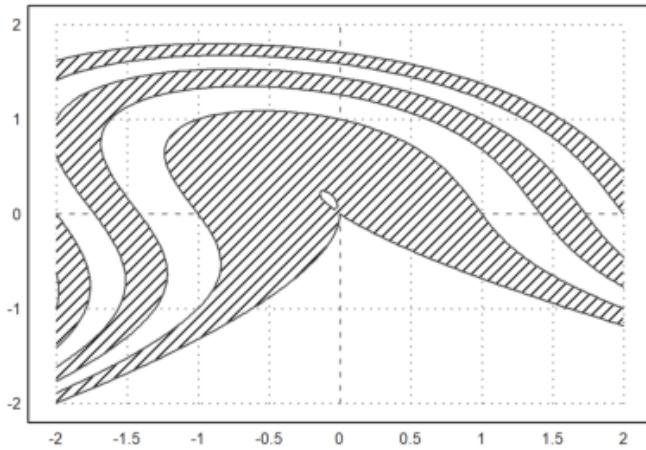


Plot implisit juga dapat menunjukkan rentang level. Maka level harus berupa matriks 2xn interval level, di mana baris pertama berisi awal dan baris kedua adalah akhir dari setiap interval. Sebagai alternatif, vektor baris sederhana dapat digunakan untuk level, dan parameter *dl* memperluas nilai level ke interval.

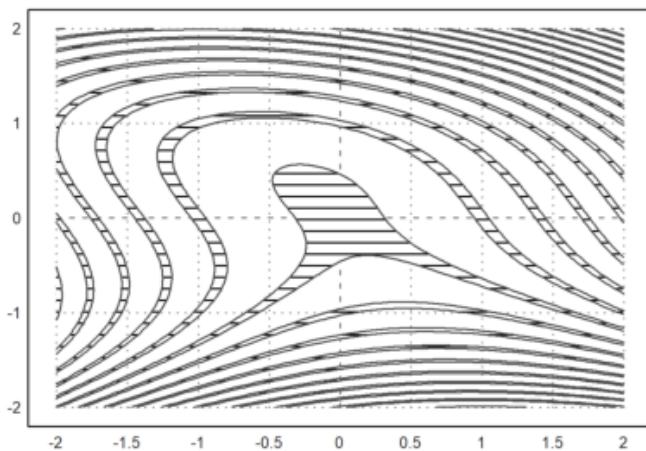
```
>plot2d("x^4+y^4",r=1.5,level=[0;1],color=blue,style="/"):
```



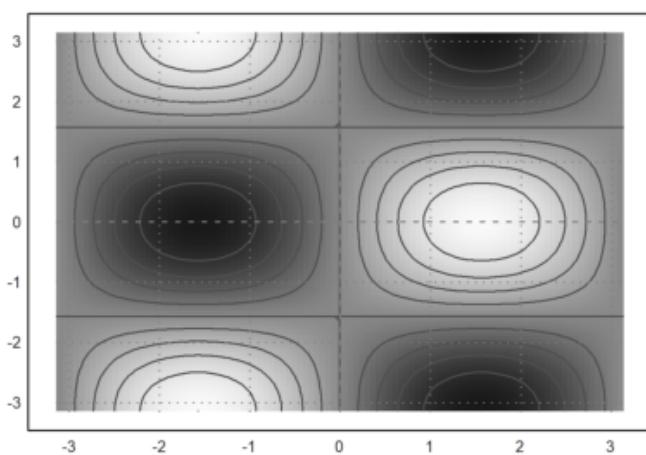
```
>plot2d("x^2+y^3+x*y",level=[0,2,4;1,3,5],style="/",r=2,n=100):
```



```
>plot2d("x^2+y^3+x*y", level=-10:20, r=2, style="-", dl=0.1, n=100) :
```



```
>plot2d("sin(x)*cos(y)", r=pi, >hue, >levels, n=100) :
```

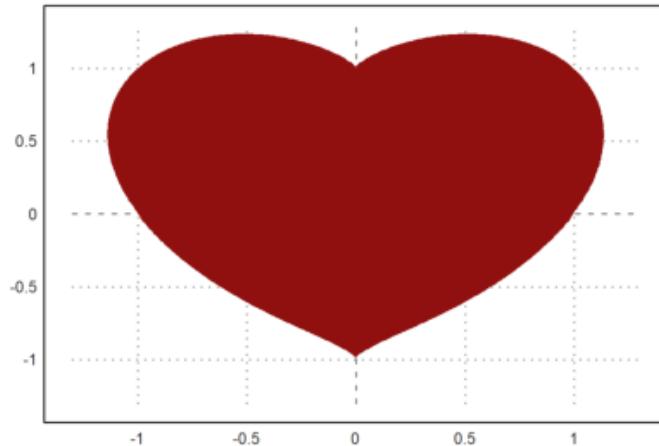


Anda juga dapat menandai suatu wilayah

$$a \leq f(x, y) \leq b.$$

Hal ini dilakukan dengan menambahkan level dengan dua baris.

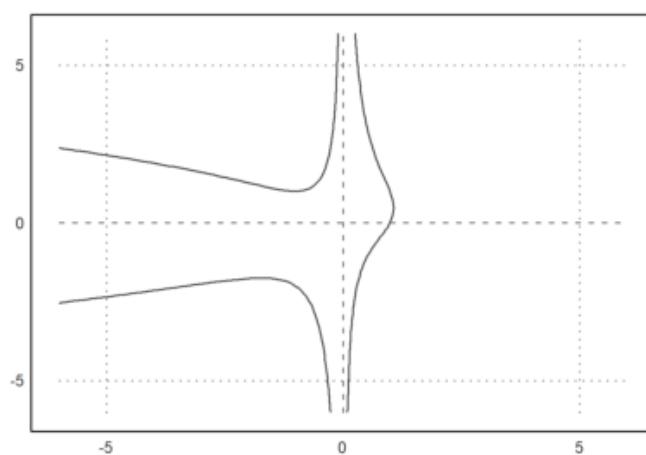
```
>plot2d("(x^2+y^2-1)^3-x^2*y^3",r=1.3, ...
>  style="#",color=red,<outline, ...
>  level=[-2;0],n=100):
```



Dimungkinkan untuk menentukan level tertentu. Misalnya, kita dapat memplot solusi dari persamaan seperti

$$x^3 - xy + x^2y^2 = 6$$

```
>plot2d("x^3-x*y+x^2*y^2",r=6,level=1,n=100):
```



```
>function starplot1 (v, style="/", color=green, lab=none) ...
```

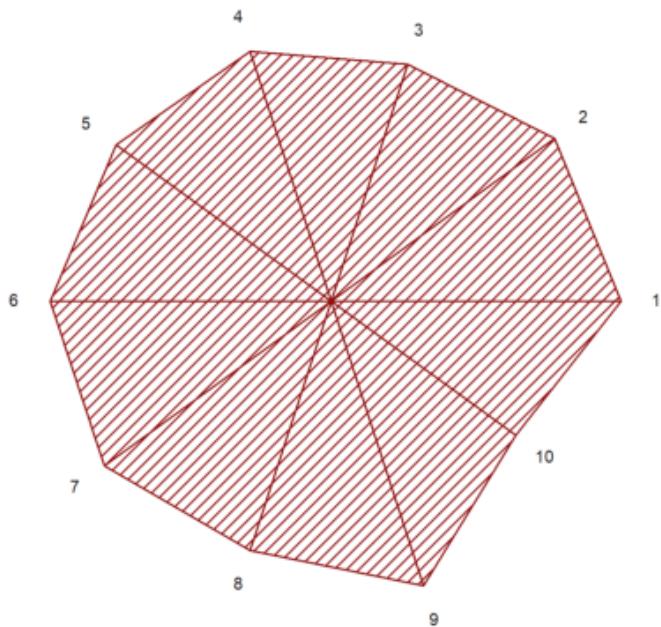
```

if !holding() then clg; endif;
w=window(); window(0,0,1024,1024);
h=holding(1);
r=max(abs(v))*1.2;
setplot(-r,r,-r,r);
n=cols(v); t=linspace(0,2pi,n);
v=v|v[1]; c=v*cos(t); s=v*sin(t);
cl=barcolor(color); st=barstyle(style);
loop 1 to n
  polygon([0,c[#],c[#+1]], [0,s[#],s[#+1]],1);
  if lab!=none then
    rlab=v[#]+r*0.1;
    {col,row}=toscreen(cos(t[#])*rlab,sin(t[#])*rlab);
    ctext(""+lab#[#],col,row-textheight()/2);
  endif;
end;
barcolor(cl); barstyle(st);
holding(h);
window(w);
endfunction

```

Tidak ada kisi-kisi atau kutu sumbu di sini. Selain itu, kami menggunakan jendela penuh untuk plot. Kami memanggil reset sebelum kami menguji plot ini untuk mengembalikan default grafis. Hal ini tidak perlu dilakukan, jika Anda yakin bahwa plot Anda berfungsi.

```
>reset; starplot1(normal(1,10)+5,color=red,lab=1:10):
```



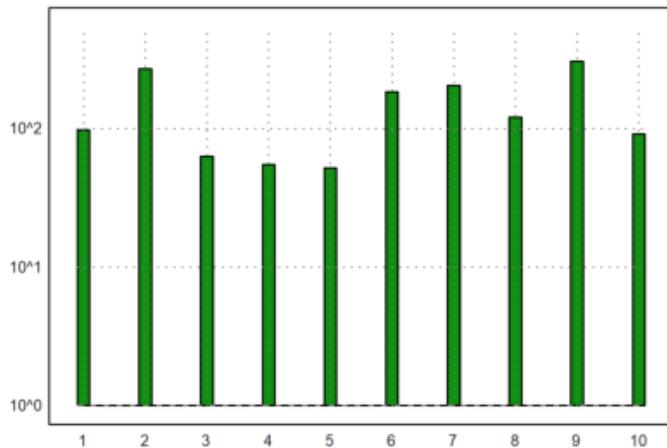
Terkadang, Anda mungkin ingin merencanakan sesuatu yang tidak dapat dilakukan oleh plot2d, tetapi ham-pir.

Pada fungsi berikut, kita melakukan plot impuls logaritmik. plot2d dapat melakukan plot logaritmik, tetapi tidak untuk batang impuls.

```
>function logimpulseplot1 (x,y) ...
{x0,y0}=makeimpulse(x,log(y)/log(10));
plot2d(x0,y0,>bar,grid=0);
h=holding(1);
frame();
xgrid(ticks(x));
p=plot();
for i=-10 to 10;
  if i<=p[4] and i>=p[3] then
    ygrid(i,yt="10^"+i);
  endif;
end;
holding(h);
endfunction
```

Mari kita uji dengan nilai yang terdistribusi secara eksponensial.

```
>aspect(1.5); x=1:10; y=-log(random(size(x)))*200; ...
>logimpulseplot1(x,y):
```



Mari kita menghidupkan kurva 2D dengan menggunakan plot langsung. Perintah plot(x,y) hanya memplot kurva ke dalam jendela plot. setplot(a,b,c,d) mengatur jendela ini.

Fungsi wait(0) memaksa plot untuk muncul pada jendela grafis. Kalau tidak, penggambaran ulang dilakukan dalam interval waktu yang jarang.

```
>function animliss (n,m) ...
```

```
t=linspace(0,2pi,500);
f=0;
c=framecolor(0);
l=linewidth(2);
setplot(-1,1,-1,1);
repeat
  clg;
  plot(sin(n*t),cos(m*t+f));
```

```

wait(0);
if testkey() then break; endif;
f=f+0.02;
end;
framecolor(c);
linewidth(l);
endfunction

```

Tekan sembarang tombol untuk menghentikan animasi ini.

```
>animliss(2,3); // lihat hasilnya, jika sudah puas, tekan ENTER
```

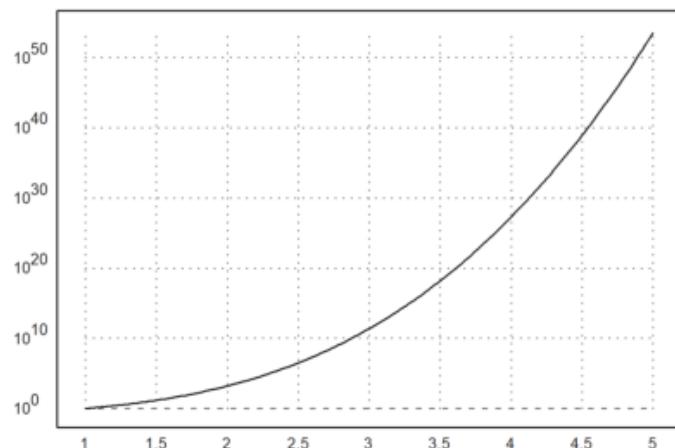
Plot Logaritmik

EMT menggunakan parameter "logplot" untuk skala logaritmik.

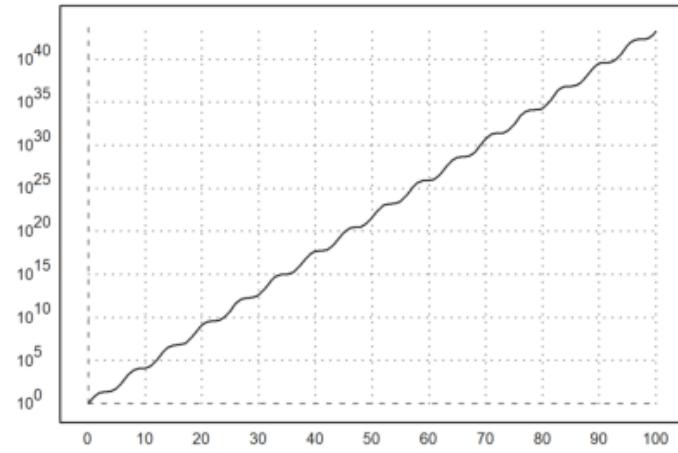
Plot logaritmik dapat diplot menggunakan skala logaritmik dalam y dengan logplot = 1, atau menggunakan skala logaritmik dalam x dan y dengan logplot = 2, atau dalam x dengan logplot = 3.

- `logplot=1`: y-logarithmic
 - `logplot=2`: x-y-logarithmic
 - `logplot=3`: x-logarithmic

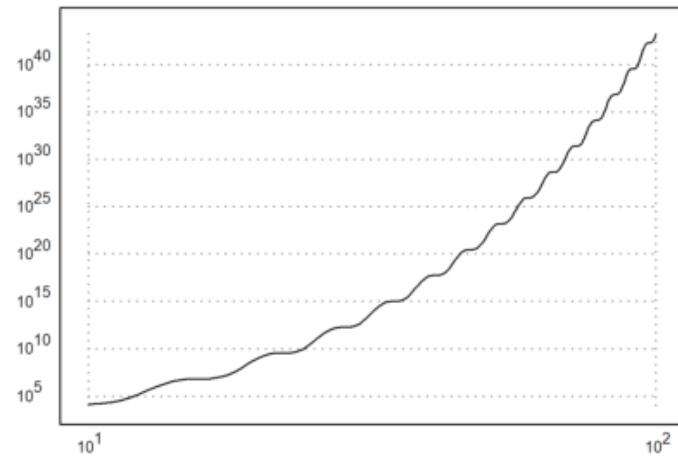
```
>plot2d("exp(x^3-x)*x^2", 1, 5, logplot=1):
```



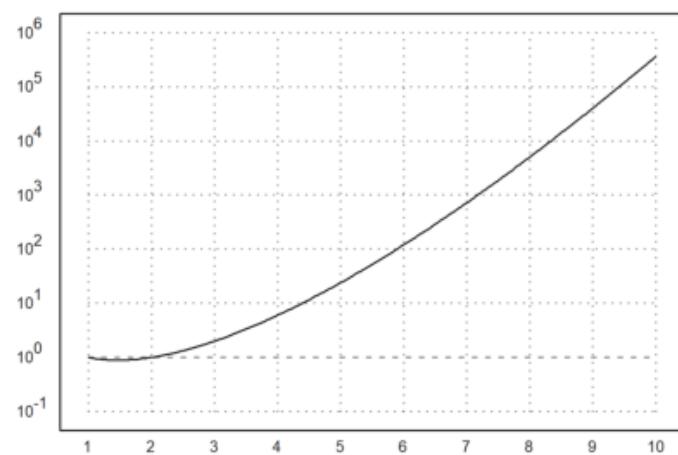
```
>plot2d("exp(x+sin(x))", 0, 100, logplot=1):
```



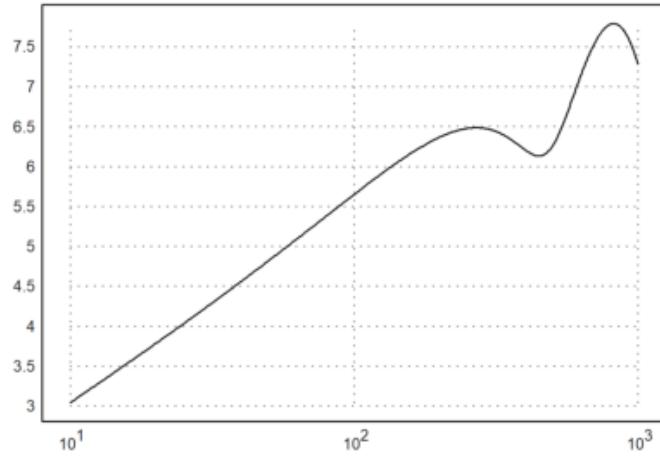
```
>plot2d("exp(x+sin(x))",10,100,logplot=2):
```



```
>plot2d("gamma(x)",1,10,logplot=1):
```

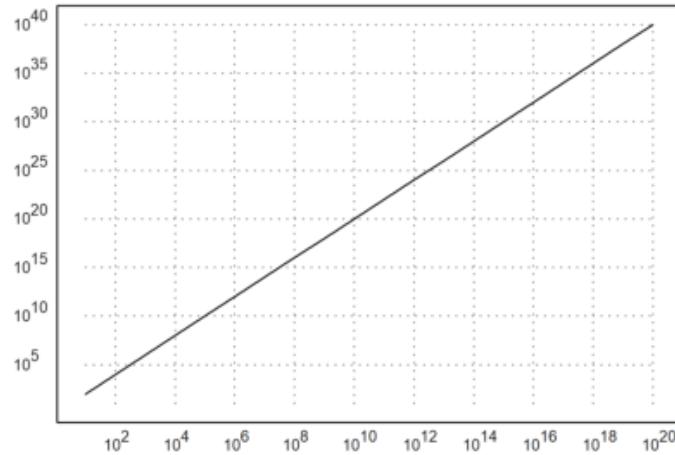


```
>plot2d("log(x*(2+sin(x/100)))",10,1000,logplot=3):
```



Hal ini juga berlaku pada plot data.

```
>x=10^(1:20); y=x^2-x;  
>plot2d(x,y,logplot=2):
```

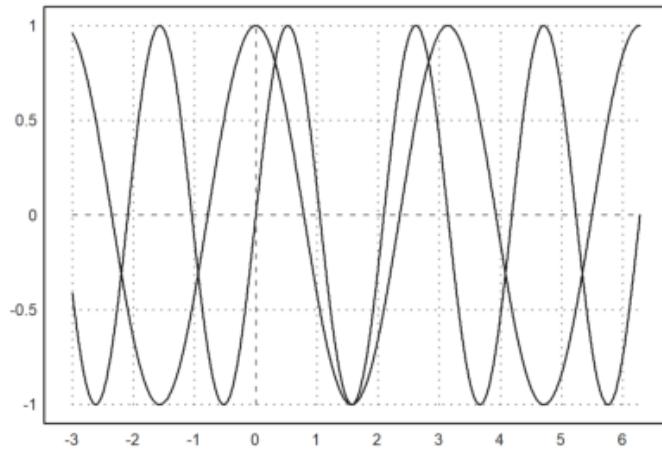


```
>
```

Contoh soal

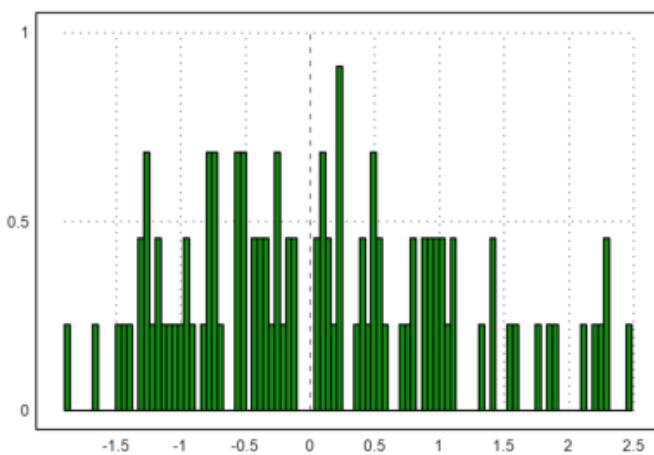
1

```
>aspect(3, 4);  
>plot2d(["sin(3*x)", "cos(2*x)"], -3, 2pi):
```



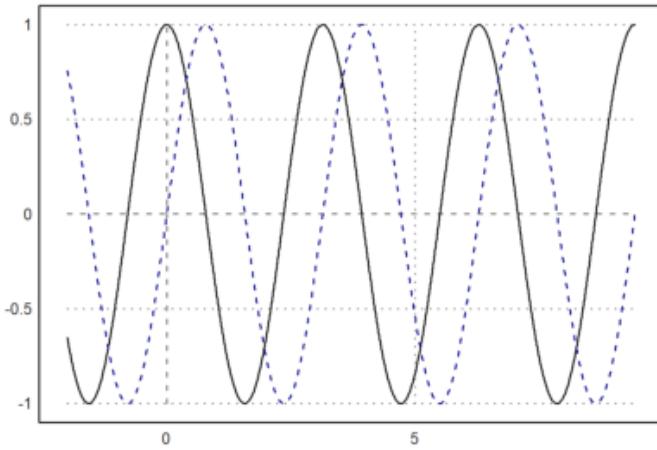
2

```
>plot2d(normal(100), distribution=100):
```



3

```
> aspect(1.5); plot2d("cos(2*x)", -2, 3pi); plot2d("sin(2*x)", color=blue, style="--", >add):
```



Rujukan Lengkap Fungsi plot2d()

```
function plot2d (xv, yv, btest, a, b, c, d, xmin, xmax, r, n, ..
logplot, grid, frame, framecolor, square, color, thickness, style, ..
auto, add, user, delta, points, addpoints, pointstyle, bar, histogram, ..
distribution, even, steps, own, adaptive, hue, level, contour, ..
nc, filled, fillcolor, outline, title, xl, yl, maps, contourcolor, ..
contourwidth, ticks, margin, clipping, cx, cy, insimg, spectral, ..
cgrid, vertical, smaller, dl, niveau, levels)
```

Multipurpose plot function for plots in the plane (2D plots). This function can do plots of functions of one variables, data plots, curves in the plane, bar plots, grids of complex numbers, and implicit plots of functions of two variables.

Parameters

x,y : equations, functions or data vectors
a,b,c,d : Plot area (default a=-2,b=2)
r : if r is set, then a=cx-r, b=cx+r, c=cy-r, d=cy+r

r can be a vector [rx,ry] or a vector [rx1,rx2,ry1,ry2].

xmin,xmax : range of the parameter for curves
auto : Determine y-range automatically (default)
square : if true, try to keep square x-y-ranges
n : number of intervals (default is adaptive)
grid : 0 = no grid and labels,

```
1 = axis only,
2 = normal grid (see below for the number of grid lines)
3 = inside axis
4 = no grid
5 = full grid including margin
6 = ticks at the frame
7 = axis only
8 = axis only, sub-ticks
```

frame : 0 = no frame

framecolor: color of the frame and the grid

margin : number between 0 and 0.4 for the margin around the plot

color : Color of curves. If this is a vector of colors,

it will be used for each row of a matrix of plots. In the case of point plots, it should be a column vector. If a row vector or a full matrix of colors is used for point plots, it will be used for each data point.

thickness : line thickness for curves

This value can be smaller than 1 for very thin lines.

style : Plot style for lines, markers, and fills.

```
For points use
"[]", "<>", ".",
"...", "...",
"*", "+", "|",
"-", "o"
"[]#", "<>#", "o#" (filled shapes)
"[]w", "<>w", "ow" (non-transparent)
For lines use
"--", "-.", ".",
".-", "-.-", "->"
For filled polygons or bar plots use
"#", "#O", "O",
"/", "\", "/",
"+", "|", "-",
t"
```

points : plot single points instead of line segments

addpoints : if true, plots line segments and points

add : add the plot to the existing plot

user : enable user interaction for functions

delta : step size for user interaction

bar : bar plot (x are the interval bounds, y the interval values)

histogram : plots the frequencies of x in n subintervals

distribution=n : plots the distribution of x with n subintervals

even : use inter values for automatic histograms.

steps : plots the function as a step function (steps=1,2)

adaptive : use adaptive plots (n is the minimal number of steps)

level : plot level lines of an implicit function of two variables

outline : draws boundary of level ranges.

If the level value is a 2xn matrix, ranges of levels will be drawn in the color using the given fill style. If outline is true, it

will be drawn in the contour color. Using this feature, regions of f(x,y) between limits can be marked.

hue : add hue color to the level plot to indicate the function

value

contour : Use level plot with automatic levels

nc : number of automatic level lines

title : plot title (default "")

xl, yl : labels for the x- and y-axis

smaller : if >0, there will be more space to the left for labels.

vertical :

Turns vertical labels on or off. This changes the global variable `verticallabels` locally for one plot. The value 1 sets only vertical text, the value 2 uses vertical numerical labels on the y axis.

`filled` : fill the plot of a curve

`fillcolor` : fill color for bar and filled curves

`outline` : boundary for filled polygons

`logplot` : set logarithmic plots

```
1 = logplot in y,  
2 = logplot in xy,  
3 = logplot in x
```

`own` :

A string, which points to an own plot routine. With `>user`, you get the same user interaction as in `plot2d`. The range will be set before each call to your function.

`maps` : map expressions (0 is faster), functions are always mapped.

`contourcolor` : color of contour lines

`contourwidth` : width of contour lines

`clipping` : toggles the clipping (default is true)

`title` :

This can be used to describe the plot. The title will appear above the plot. Moreover, a label for the x and y axis can be added with `xl="string"` or `yl="string"`. Other labels can be added with the functions `label()` or `labelbox()`. The title can be a unicode string or an image of a Latex formula.

`cgrid` :

Determines the number of grid lines for plots of complex grids. Should be a divisor of the the matrix size minus 1 (number of subintervals). `cgrid` can be a vector `[cx,cy]`.

Overview

The function can plot

- expressions, call collections or functions of one variable,
- parametric curves,
- x data against y data,
- implicit functions,
- bar plots,
- complex grids,
- polygons.

If a function or expression for `xv` is given, `plot2d()` will compute values in the given range using the function or expression. The expression must be an expression in the variable `x`. The range must be defined in the parameters `a` and `b` unless the default range should be used. The y-range will be computed automatically, unless `c` and `d` are specified, or a radius `r`, which yields the range `r,r` for `x` and `y`. For plots of functions, `plot2d` will use an

adaptive evaluation of the function by default. To speed up the plot for complicated functions, switch this off with <adaptive, and optionally decrease the number of intervals n. Moreover, plot2d() will by default use mapping. I.e., it will compute the plot element for element. If your expression or your functions can handle a vector x, you can switch that off with <maps for faster evaluation. Note that adaptive plots are always computed element for element. If functions or expressions for both xv and for yv are specified, plot2d() will compute a curve with the xv values as x-coordinates and the yv values as y-coordinates. In this case, a range should be defined for the parameter using xmin, xmax. Expressions contained in strings must always be expressions in the parameter variable x.

BAB 4

KB PEKAN 5: MENGGUNAKAN EMT UNTUK MENGAMBAR GRAFIK 3 DIMENSI (3D)

article
eumat

Shalih Abdillah(22305144009)

Menggambar Plot 3D dengan EMT

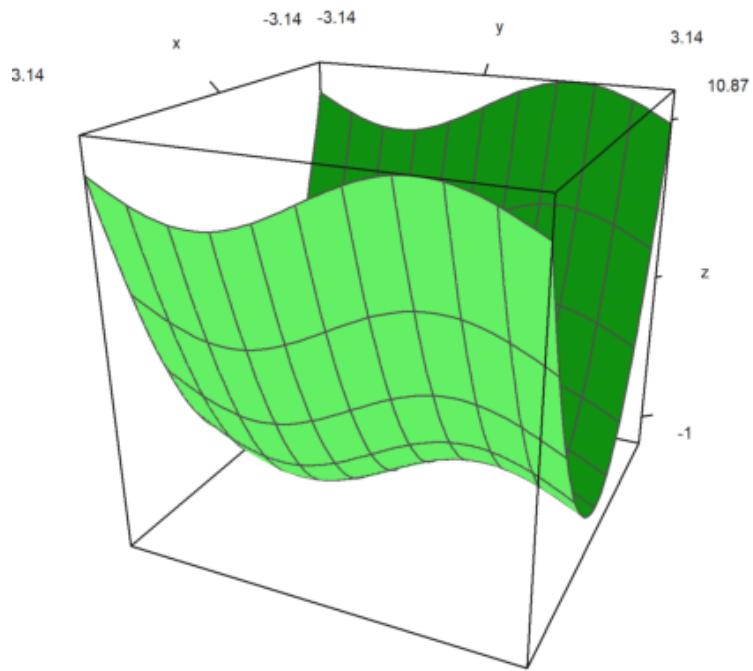
Bagian ini merupakan pengenalan plot 3d pada Euler. Kita memerlukan plot 3D untuk memvisualisasikan dari suatu fungsi dua variabel.

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

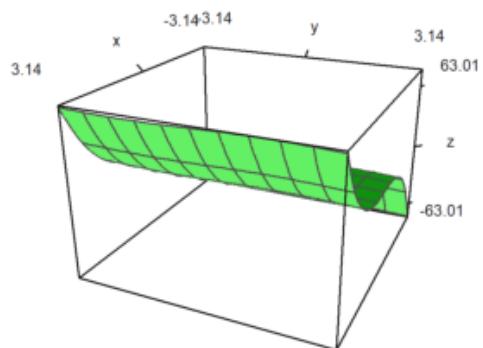
- permukaan dengan bayangan dan garis level atau rentang level,
- awan poin,
- kurva parametrik,
- permukaan implisit.

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); plot3d("x^2+sin(y)",r=pi):
```

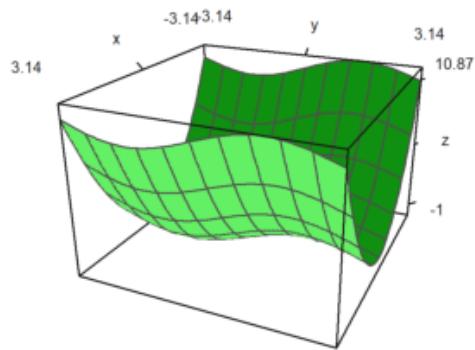


```
>aspect(3,2); plot3d("2*x^3+sin(y)",r=pi):
```

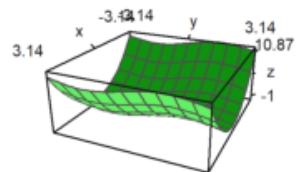


Dibawah ini percobaan dari beberapa perintah dengan mengganti salah satu dari variabelnya.

```
>aspect(1.5); plot3d("x^2+sin(y)",r=pi):
```



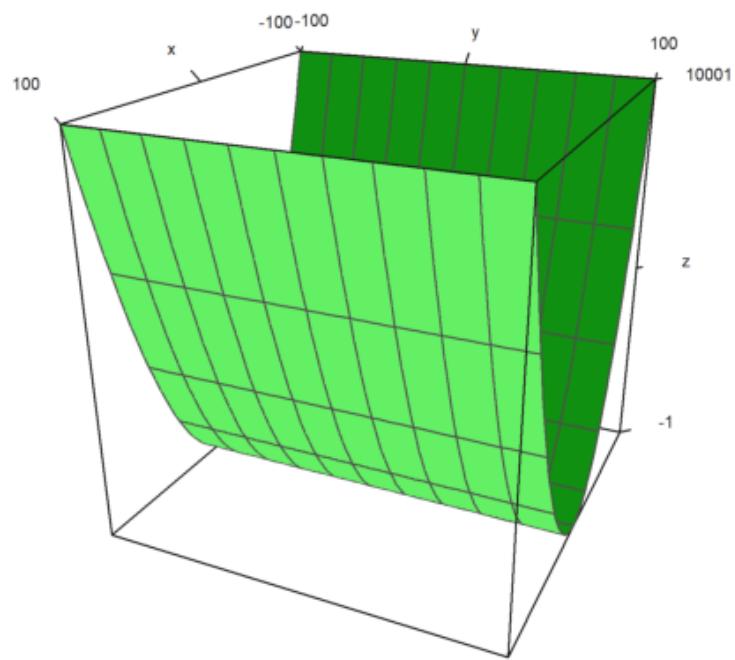
```
>aspect(2.5); plot3d("x^2+sin(y)",r=pi):
```



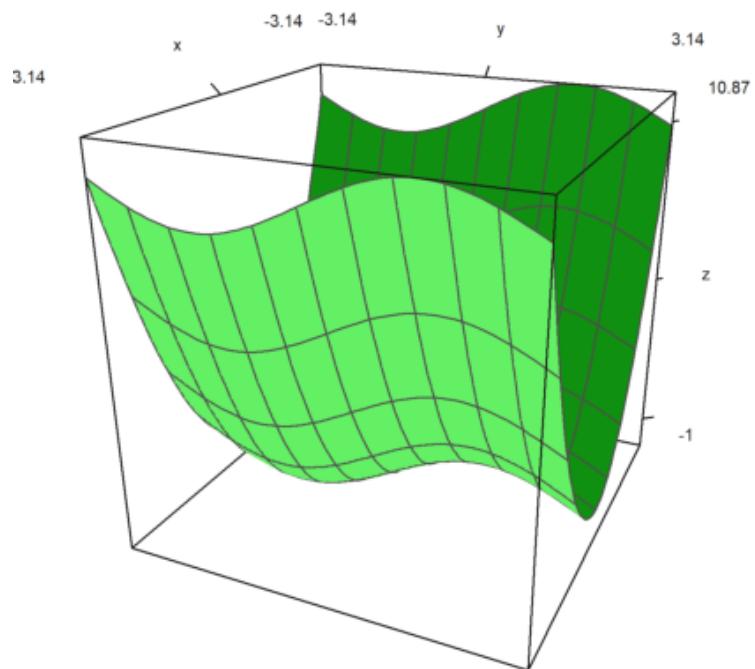
Dari berbagai percobaan diatas, dapat dilihat bahwa semakin besar nilai aspect-nya, hasil plot 3D semakin kecil.

Aspect ialah proporsi antara sumbu x dan sumbu y. Artinya jika aspect 1 maka proporsi x dan y sama. Selanjutnya akan dilihat perbedaan hasil grafik dengan parameter yang berubah adalah r.

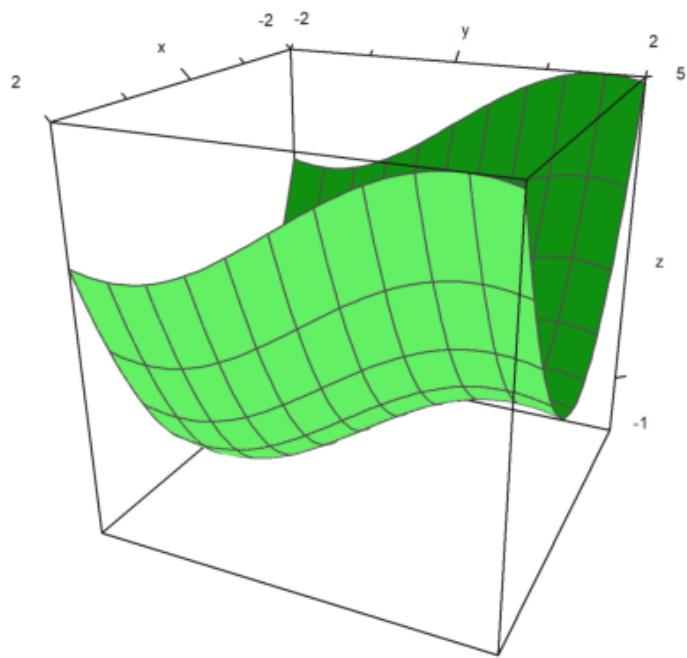
```
>aspect(1); plot3d("x^2+sin(y)",r=100):
```



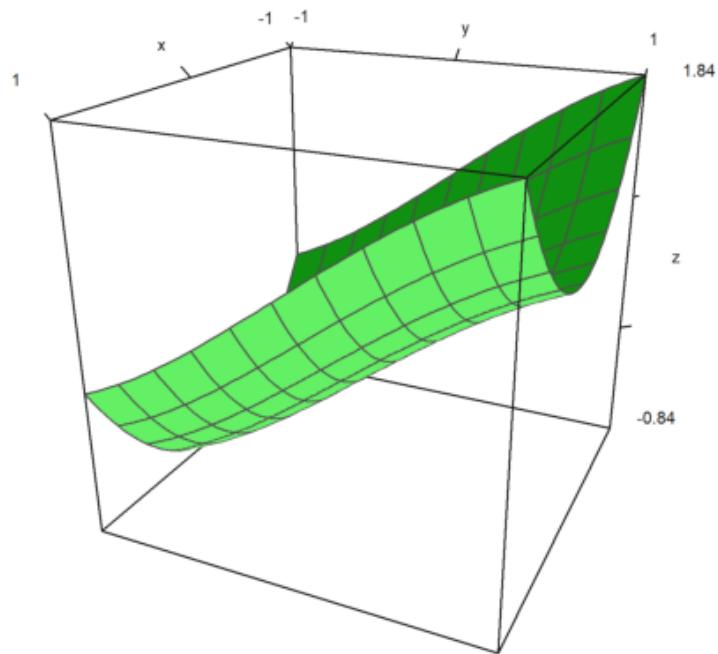
```
>aspect(1); plot3d("x^2+sin(y)",r=pi):
```



```
>aspect(1); plot3d("x^2+sin(y)",r=2):
```



```
>aspect(1); plot3d("x^2+sin(y)",r=1):
```



Berbagai perbedaan hasil yang diperoleh ketika nilai r nya dirubah. Semakin besar nilai r nya plot yang dihasilkan semakin kecil sehingga ketelitiannya semakin kecil. Di sisi lain jika nilai r yang ditetapkan terlalu kecil, plot yang dihasilkan juga kurang baik karena belum dapat mewakili plot secara garis beras. Darisini diperoleh informasi bahwasanya pemilihan nilai r haruslah tepat agar plot yang dihasilkan lebih akurat dan memahamkan, sesuai yang kita butuhkan.

Fungsi Dua Variabel

Untuk grafik fungsi, gunakan

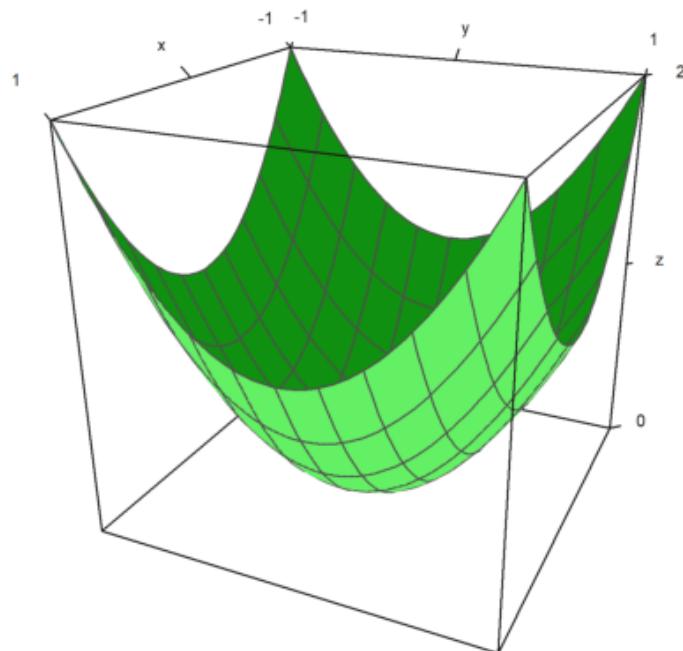
- ekspresi sederhana dalam x dan y,
- nama fungsi dari dua variabel
- atau matriks data.

Standarnya adalah kotak kawat yang diisi dengan warna berbeda di kedua sisi. Perhatikan bahwa jumlah default interval grid adalah 10, tetapi plot menggunakan jumlah default 40x40 persegi panjang untuk membangun permukaan. Ini bisa diubah.

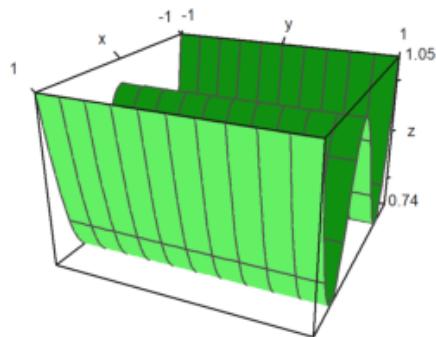
- n=40, n=[40,40]: jumlah garis grid di setiap arah
- grid=10, grid=[10,10]: jumlah garis grid di setiap arah.

Kami menggunakan default n=40 dan grid=10.

```
>plot3d ("x^2+y^2") :
```



```
>plot3d ("x^2+cos (x)^5") :
```

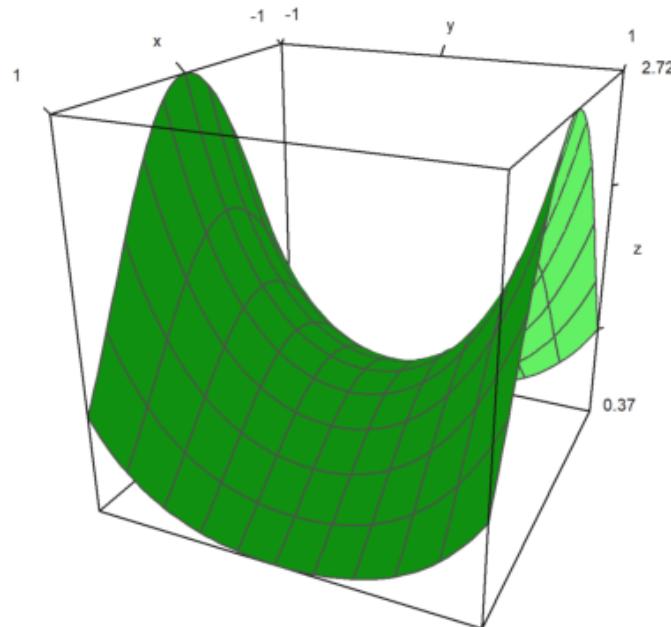


Interaksi pengguna dimungkinkan dengan >parameter pengguna. Pengguna dapat menekan tombol berikut.

- kiri, kanan, atas, bawah: putar sudut pandang
- +,-: memperbesar atau memperkecil
- a: menghasilkan anaglyph (lihat di bawah)
- l: beralih memutar sumber cahaya (lihat di bawah)
- spasi: reset ke default
- kembali: akhiri interaksi

```
>plot3d("exp(-x^2+y^2)",>user, ...
> title="Turn with the vector keys (press return to finish)":
```

Turn with the vector keys (press return to finish)



Rentang plot untuk fungsi dapat ditentukan dengan

- a,b: rentang-x
- c,d: rentang-y

- r: persegi simetris di sekitar (0,0).

- n: jumlah subinterval untuk plot.

Ada beberapa parameter untuk menskalakan fungsi atau mengubah tampilan grafik.

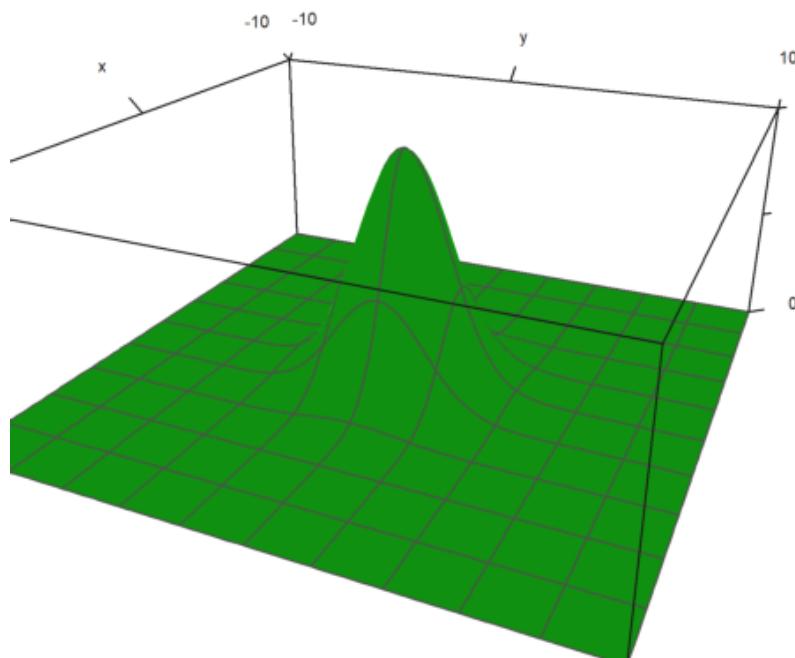
fscale: skala ke nilai fungsi (defaultnya adalah <fscale>).

skala: angka atau vektor 1x2 untuk skala ke arah x dan y.

bingkai: jenis bingkai (default 1).

Catatan: Jika sudah menggunakan atau mendefinisikan nilai r maka tidak perlu mendefinisikan lagi nilai a,b,c,dan d nya.

```
>plot3d("exp(-(x^2+y^2)/5)",r=10,n=90,fscale=4,scale=1.5,frame=1):
```



Tampilan dapat diubah dengan berbagai cara.

- jarak: jarak pandang ke plot.

- zoom: nilai zoom.

- sudut: sudut terhadap sumbu y negatif dalam radian.

- tinggi: ketinggian tampilan dalam radian.

Nilai default dapat diperiksa atau diubah dengan fungsi view(). Ini mengembalikan parameter dalam urutan di atas.

```
>view
```

```
[5, 2.6, 2, 0.4]
```

5 menunjukkan jarak pandang ke plot,

2.6 menunjukkan nilai zoom,

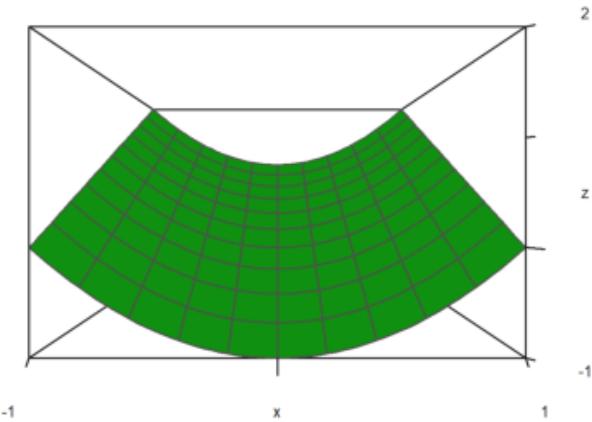
2 menunjukkan nilai sudut terhadap y negatif dalam radian, dan

0.4 menunjukkan ketinggian tampilan dalam radian.

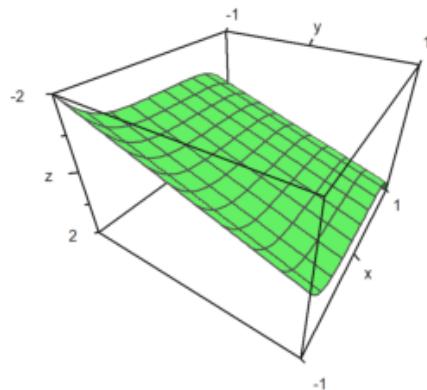
Jarak yang lebih dekat membutuhkan lebih sedikit zoom. Efeknya lebih seperti lensa sudut lebar.

Dalam contoh berikut, sudut=0 dan tinggi=0 terlihat dari sumbu y negatif. Label sumbu untuk y disembunyikan dalam kasus ini.

```
>plot3d("x^2+y",distance=3,zoom=2,angle=0,height=0):
```

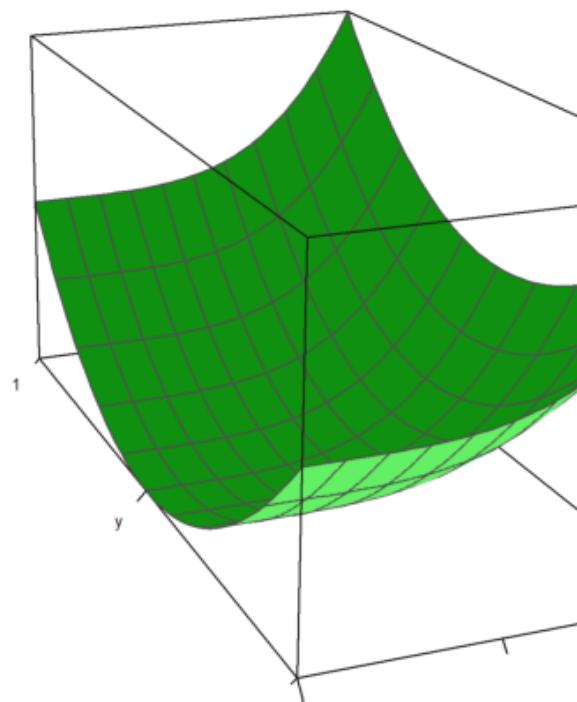


```
>plot3d("x^5+y",distance=3,zoom=1.4,angle=1,height=10):
```



Plot terlihat selalu ke pusat kubus plot. Anda dapat memindahkan pusat dengan parameter tengah.

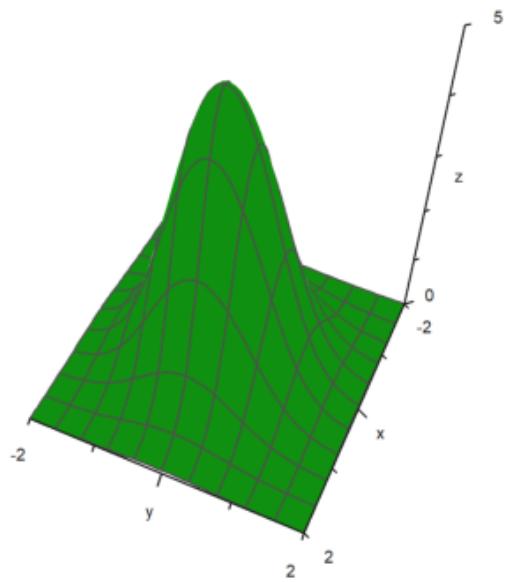
```
>plot3d("x^4+y^2",a=0,b=1,c=-1,d=1,angle=-20°,height=20°, ...
> center=[0.4,0,0],zoom=5):
```



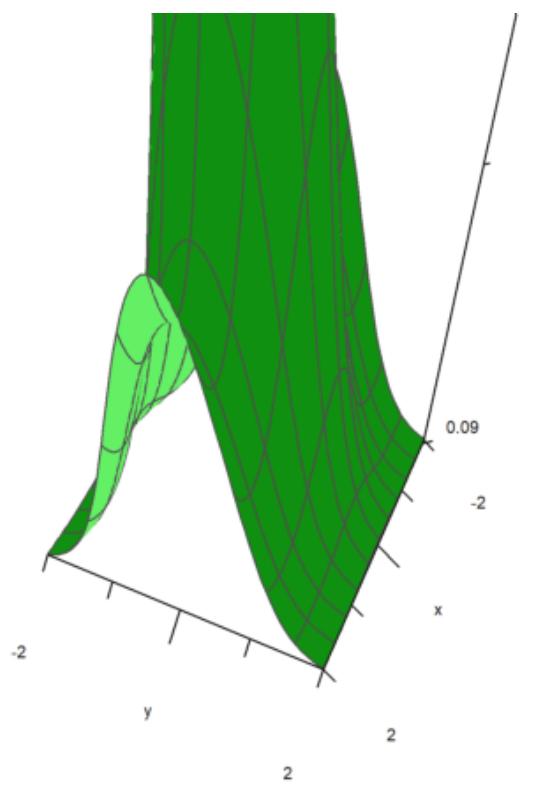
Plot diskalakan agar sesuai dengan kubus satuan untuk dilihat. Jadi tidak perlu mengubah jarak atau zoom tergantung pada ukuran plot. Namun, label mengacu pada ukuran sebenarnya.

Jika Anda mematikannya dengan `scale=false`, Anda perlu berhati-hati, bahwa plot masih cocok dengan jendela plot, dengan mengubah jarak pandang atau zoom, dan memindahkan pusat.

```
>plot3d("5*exp(-x^2-y^2)",r=2,<fscale,<scale,distance=13,height=50°, ...
>  center=[0,0,-2],frame=3):
```

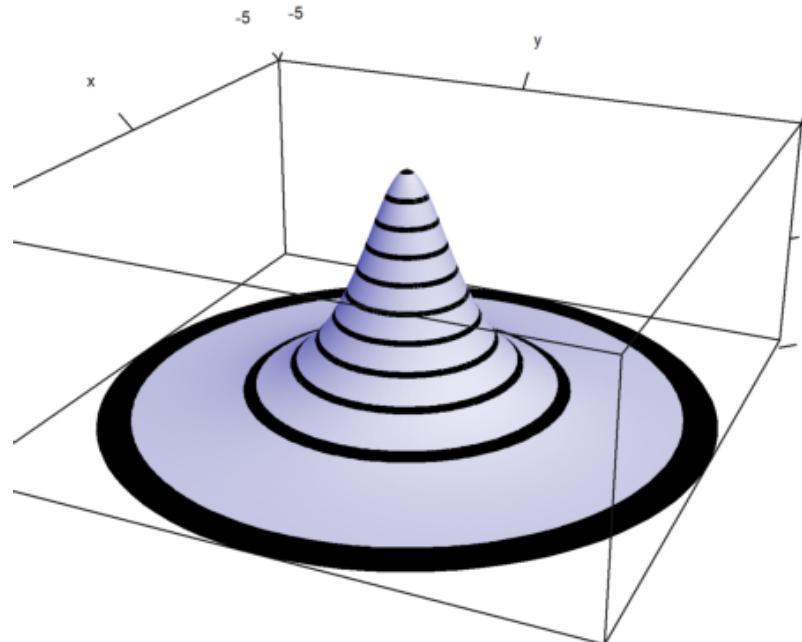


```
>plot3d("5*exp(cos(x)^2-y^2)",r=2,<fscale,<scale,distance=13,height=50°, ...
> center=[0,0,-2],frame=3):
```

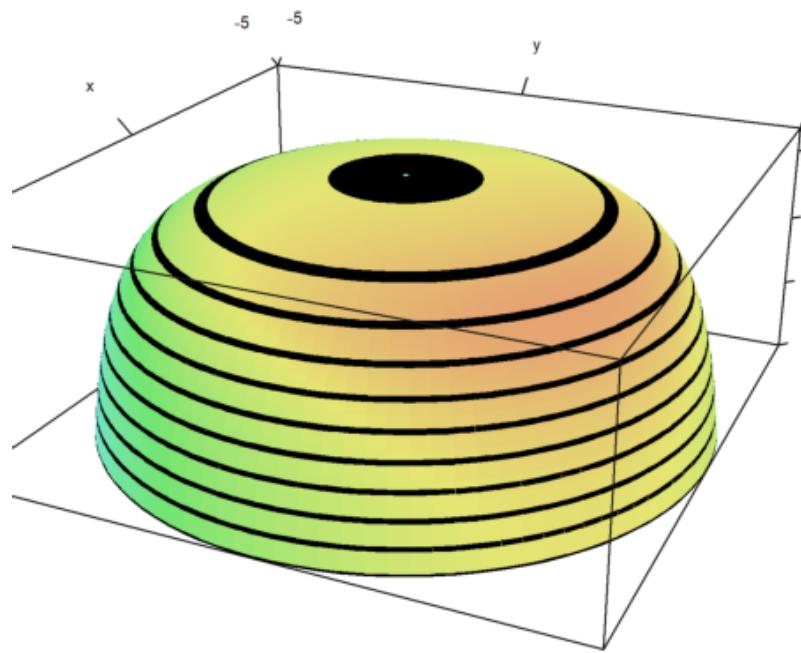


Sebuah plot kutub juga tersedia. Parameter polar=true menggambar plot polar. Fungsi tersebut harus tetap merupakan fungsi dari x dan y. Parameter "fscale" menskalakan fungsi dengan skala sendiri. Jika tidak, fungsi diskalakan agar sesuai dengan kubus.

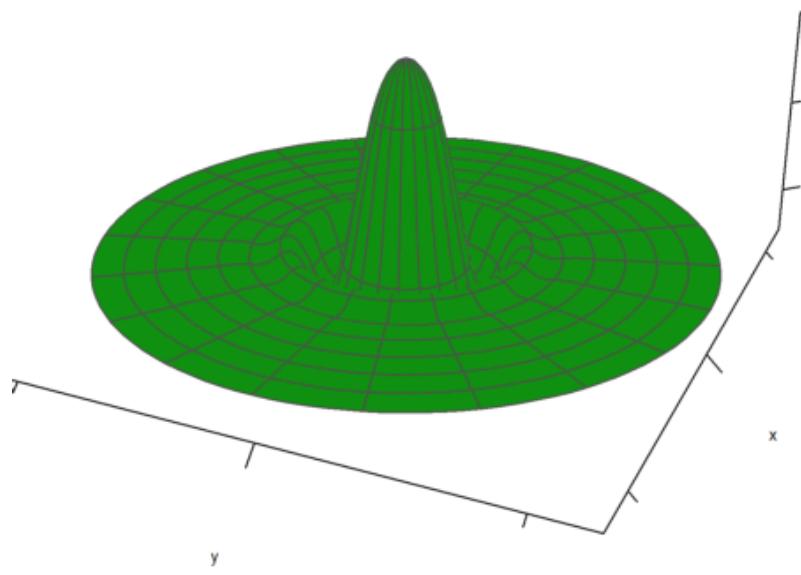
```
>plot3d("1/(x^2+y^2+1)",r=5,>polar, ...
>fscale=2,>hue,n=100,zoom=4,>contour,color=blue):
```



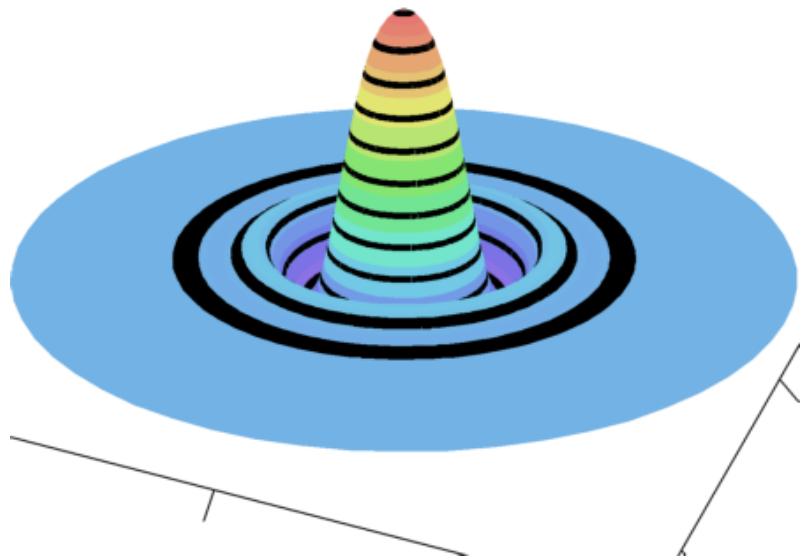
```
>plot3d("1/(x^2+y^2-30)",r=5,>polar, ...
>fscale=2,>hue,n=100,zoom=4,>contour,color=white):
```



```
>function f(r) := exp(-r/2)*cos(r); ...
>plot3d("f(x^2+y^2)",>polar,scale=[1,1,0.4],r=2pi,frame=3,zoom=4):
```



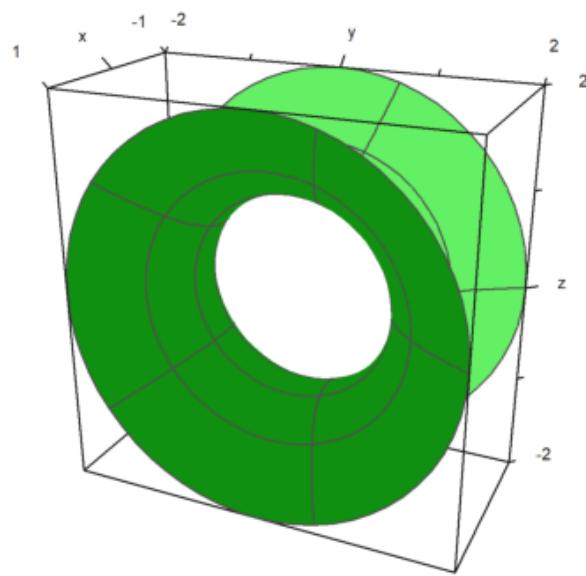
```
>function f(r) := exp(-r/2)*cos(r); ...
>plot3d("f(x^2+y^2)",>polar,scale=[1,1,0.4],r=2pi,frame=3,zoom=5,>contour,color=white):
```



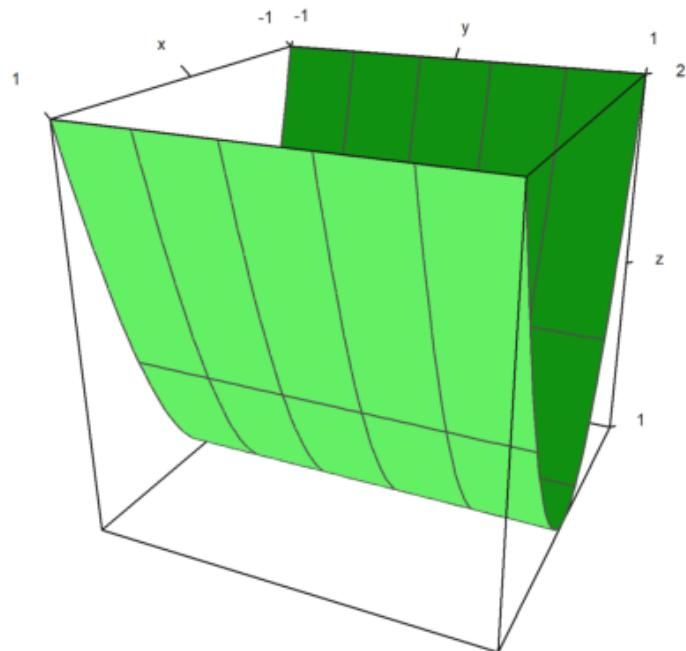
Rotasi parameter memutar fungsi dalam x di sekitar sumbu x.

- rotate=1: Menggunakan sumbu x
- rotate=2: Menggunakan sumbu z

```
>plot3d("x^2+1",a=-1,b=1,rotate=true,grid=5):
```

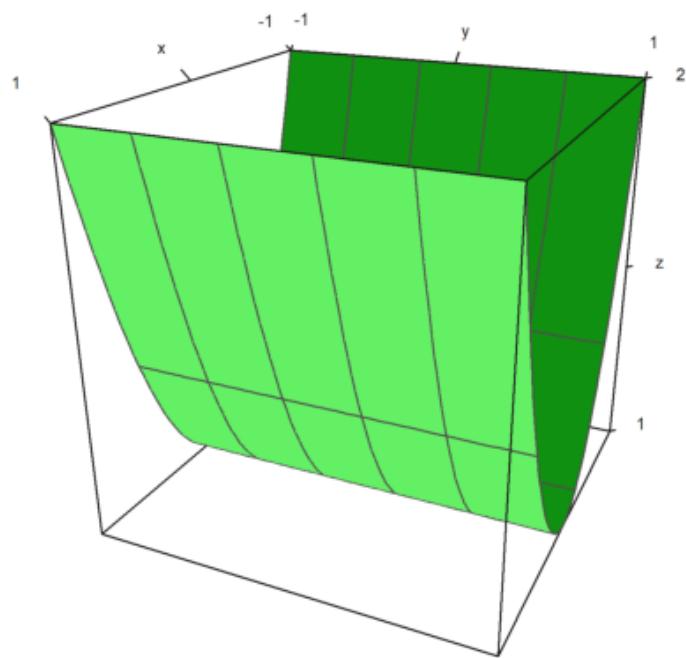


```
>plot3d("x^2+1",a=-1,b=1,rotate=false,grid=5):
```



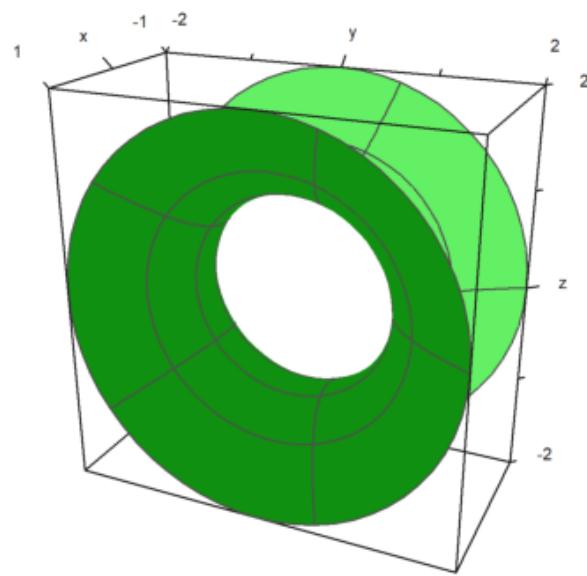
Perintah `rotate = false` menghasilkan plot yang sama dengan perintah tanpa `rotate`.

```
>plot3d("x^2+1",a=-1,b=1,grid=5):
```



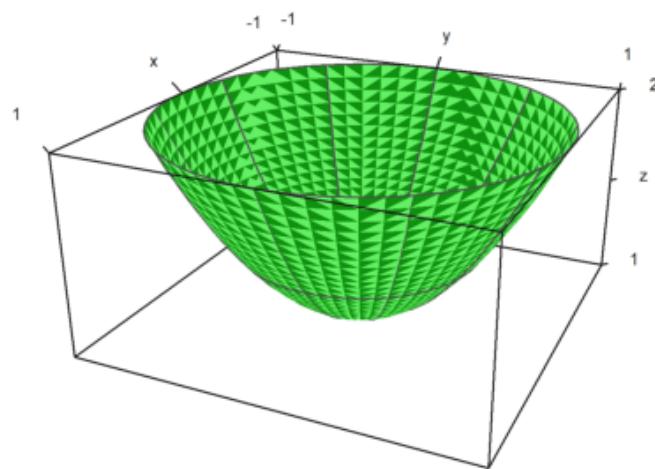
Dibawah ini merupakan hasil plot dari dngsi yang di rotasi dengan perintah rotate = 1 artinya dirotasi terhadap sumbu y, yang menghasilkan dengan nilai yang sama dengan perintah rotate = true.

```
>plot3d("x^2+1",a=-1,b=1,rotate=1,grid=5):
```



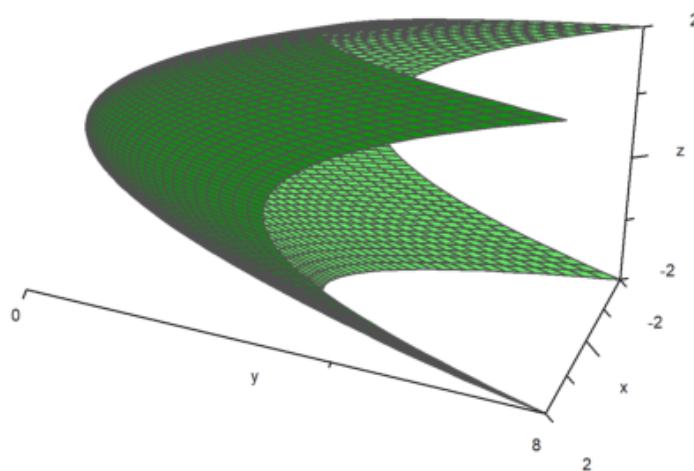
Dibawah ini merupakan hasil plot dari dngsi yang di rotasi dengan perintah rotate = 2 artinya dirotasi terhadap sumbu z.

```
>plot3d("x^2+1",a=-1,b=1,rotate=2,grid=5):
```



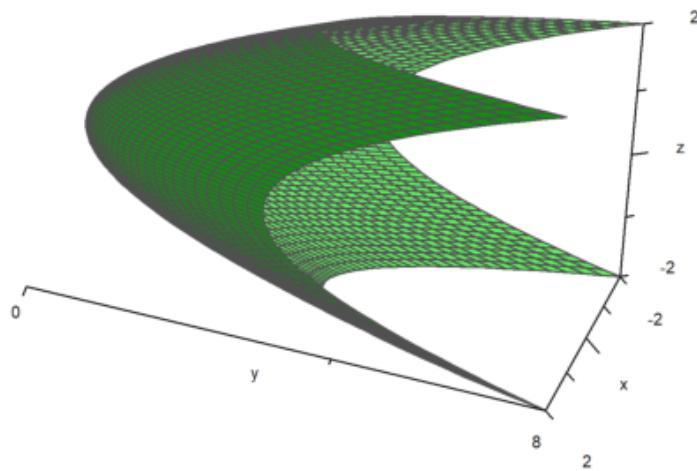
Dibawah ini merupakan plot dengan tiga fungsi.

```
>plot3d("x", "x^2+y^2", "y", r=2, zoom=3.5, frame=3) :
```



Dibawah ini dilakukan percobaan untuk plot tiga fungsi, ternyata tidak bisa dikenai perintah rotate (rotasi).

```
>plot3d("x", "x^2+y^2", "y", r=2, zoom=3.5, frame=3) :
```



Plot Kontur

Untuk plot, Euler menambahkan garis grid. Sebagai gantinya dimungkinkan untuk menggunakan garis level dan rona satu warna atau rona berwarna spektral. Euler dapat menggambar tinggi fungsi pada plot dengan bayangan. Di semua plot 3D, Euler dapat menghasilkan anaglyph merah/sian.

-> hue: Menyalakan bayangan cahaya alih-alih kabel.

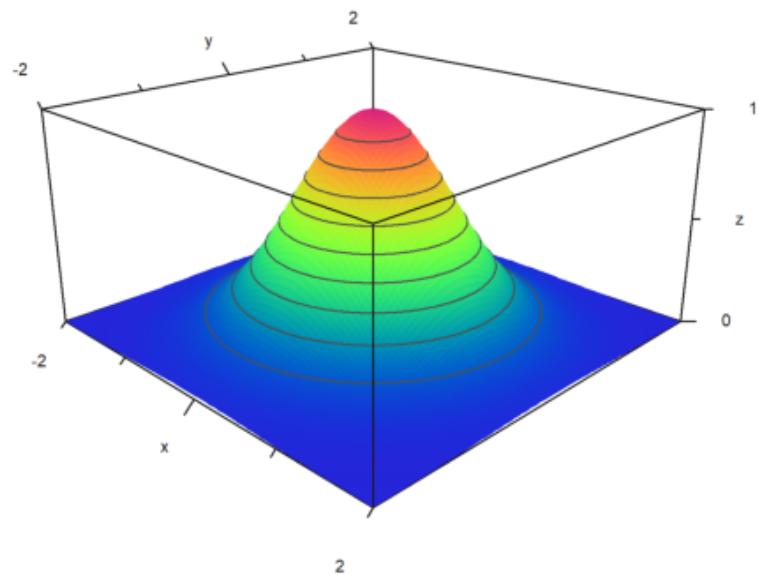
-> kontur: Memplot garis kontur otomatis pada plot.

- level=... (atau level): Sebuah vektor nilai untuk garis kontur.

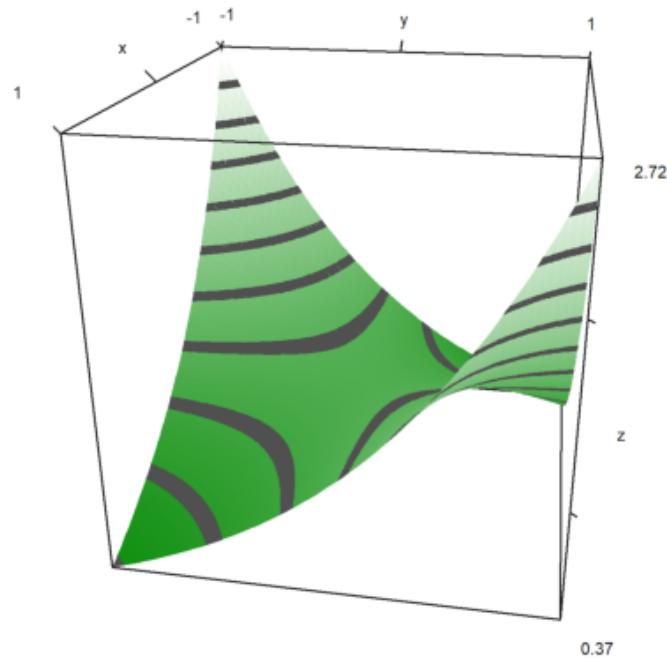
Standarnya adalah level="auto", yang menghitung beberapa garis level secara otomatis. Seperti yang Anda lihat di plot, level sebenarnya adalah rentang level.

Gaya default dapat diubah. Untuk plot kontur berikut, kami menggunakan grid yang lebih halus untuk 100x100 poin, skala fungsi dan plot, dan menggunakan sudut pandang yang berbeda.

```
>plot3d("exp(-x^2-y^2)",r=2,n=100,level="thin", ...
> >contour,>spectral,fscale=1,scale=1.1,angle=45°,height=20°) :
```



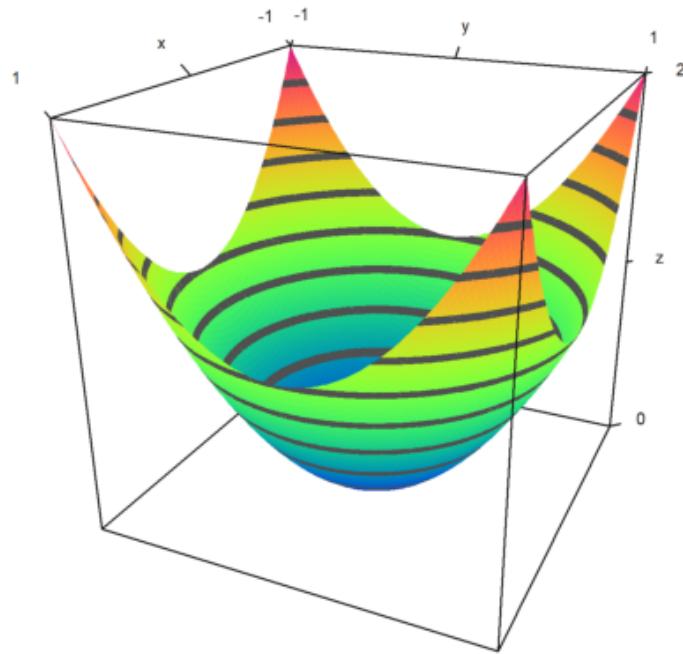
```
>plot3d("exp(x*y)",angle=100°,>contour,color=green):
```



Bayangan default menggunakan warna abu-abu. Tetapi rentang warna spektral juga tersedia.
 -> spektral: Menggunakan skema spektral default
 - color=...: Menggunakan warna khusus atau skema spektral

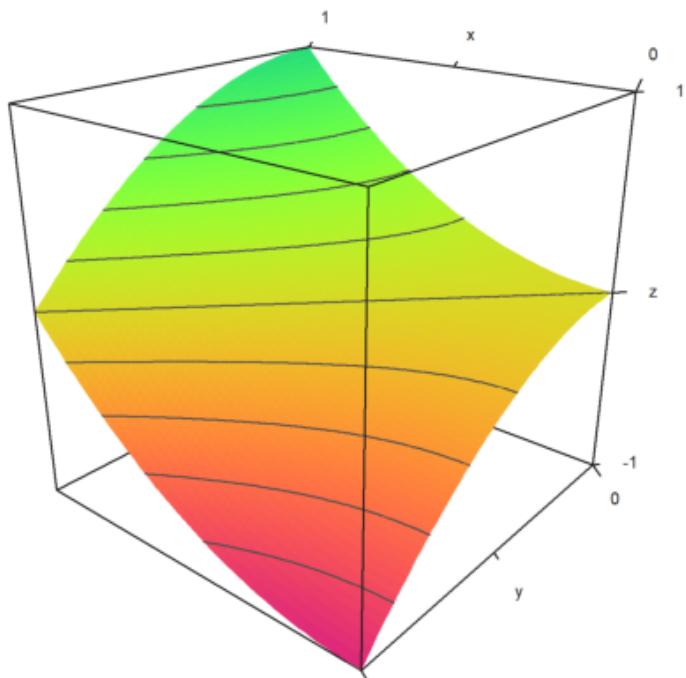
Untuk plot berikut, kami menggunakan skema spektral default dan menambah jumlah titik untuk mendapatkan tampilan yang sangat halus.

```
>plot3d("x^2+y^2",>spectral,>contour,n=100):
```



Alih-alih garis level otomatis, kita juga dapat mengatur nilai garis level. Ini akan menghasilkan garis level tipis alih-alih rentang level.

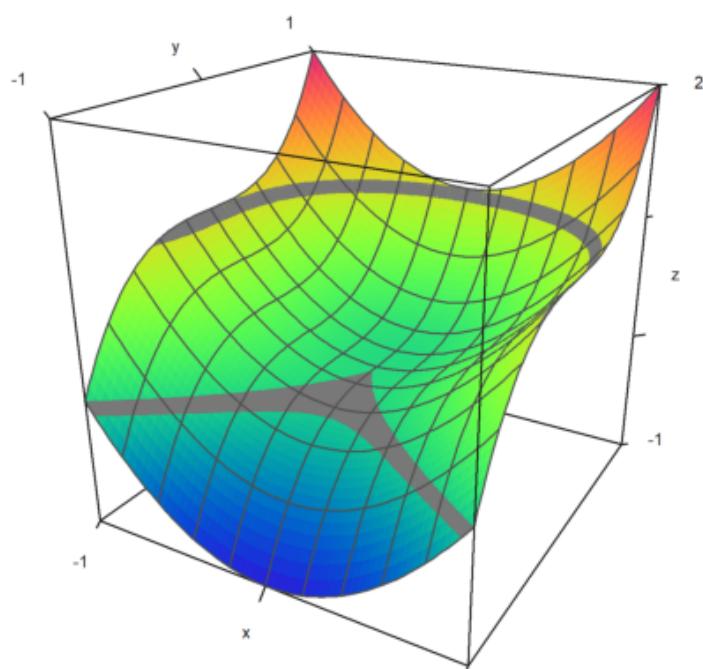
```
>plot3d("x^2-y^2",0,1,0,1,angle=220°,level=-1:0.2:1,color=redgreen):
```



Dalam plot berikut, kami menggunakan dua pita level yang sangat luas dari -0,1 hingga 1, dan dari 0,9 hingga 1. Ini dimasukkan sebagai matriks dengan batas level sebagai kolom.

Selain itu, kami melapisi kisi dengan 10 interval di setiap arah.

```
>plot3d("x^2+y^3",level=[-0.1,0.9;0,1], ...
> >spectral,angle=30°,grid=10,contourcolor=gray):
```

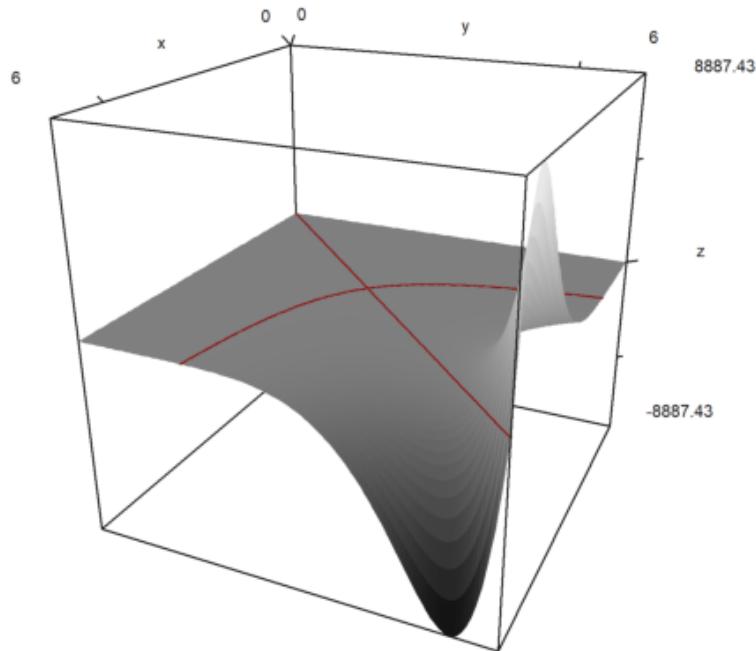


Dalam contoh berikut, kami memplot himpunan, di mana

$$f(x, y) = x^y - y^x = 0$$

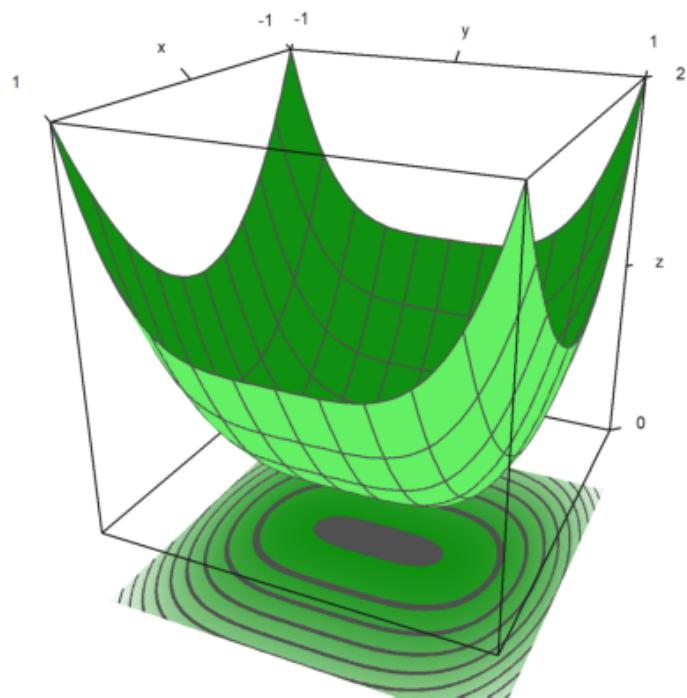
Kami menggunakan satu garis tipis untuk garis level.

```
>plot3d("x^y-y^x", level=0, a=0, b=6, c=0, d=6, contourcolor=red, n=100) :
```



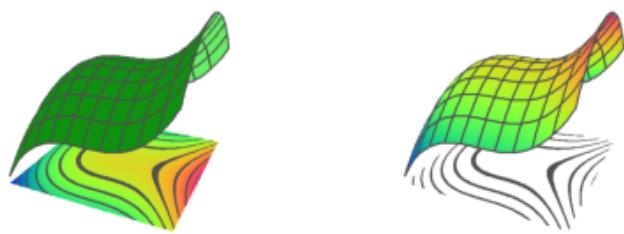
Dimungkinkan untuk menunjukkan bidang kontur di bawah plot. Warna dan jarak ke plot dapat ditentukan.

```
>plot3d("x^2+y^4", >cp, cpcolor=green, cpdelta=0.2) :
```



Berikut adalah beberapa gaya lagi. Kami selalu mematikan frame, dan menggunakan berbagai skema warna untuk plot dan grid.

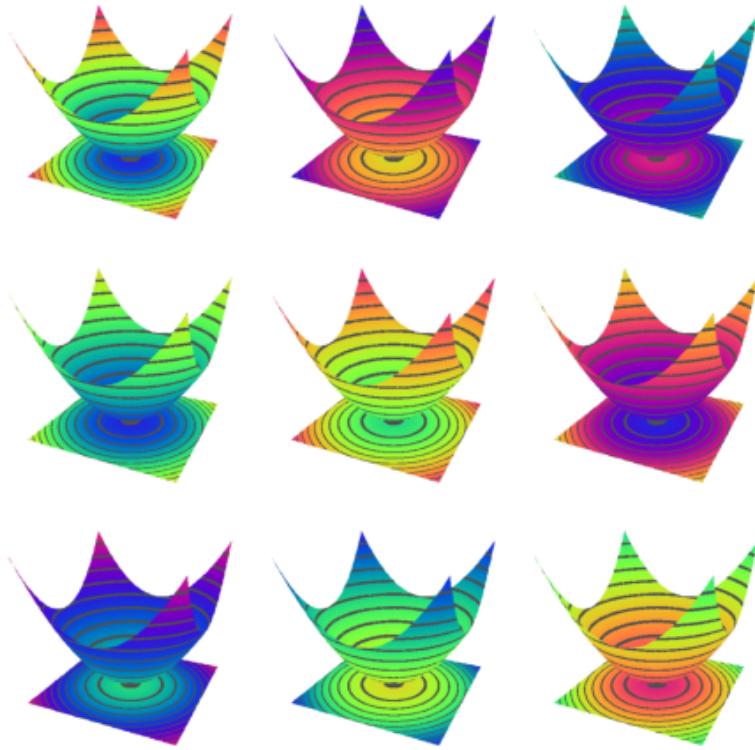
```
>figure(2,2); ...
>expr="y^3-x^2"; ...
>figure(1); ...
> plot3d(expr,<frame,>cp,cpcolor=spectral); ...
>figure(2); ...
> plot3d(expr,<frame,>spectral,grid=10,cp=2); ...
>figure(3); ...
> plot3d(expr,<frame,>contour,color=gray,nc=5,cp=3,cpcolor=greenred); ...
>figure(4); ...
> plot3d(expr,<frame,>hue,grid=10,>transparent,>cp,cpcolor=gray); ...
>figure(0):
```



Ada beberapa skema spektral lainnya, bernomor dari 1 hingga 9. Tetapi Anda juga dapat menggunakan warna=nilai, di mana nilai

- spektral: untuk rentang dari biru ke merah
- putih: untuk rentang yang lebih redup
- kuningbiru, ungu hijau, birukuning, hijaumerah
- biru kuning, 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):
```



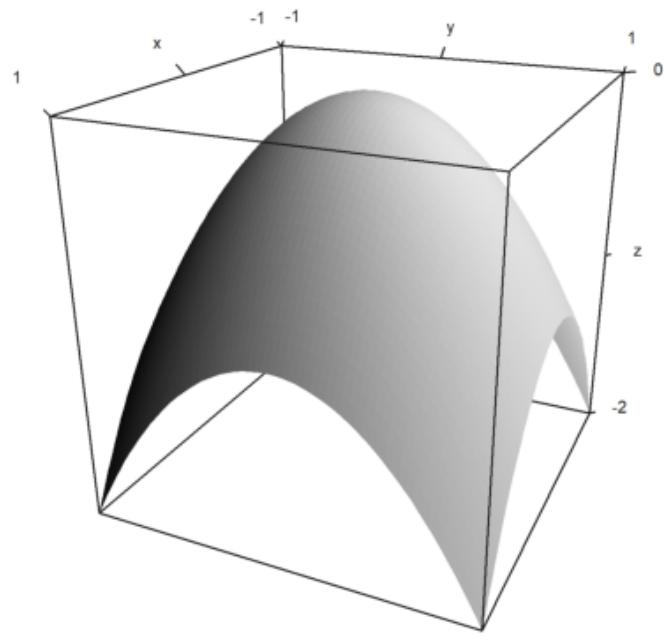
Sumber cahaya dapat diubah dengan l dan tombol cursor selama interaksi pengguna. Itu juga dapat diatur dengan parameter.

- 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)":
```

Press I and cursor keys (return to exit)



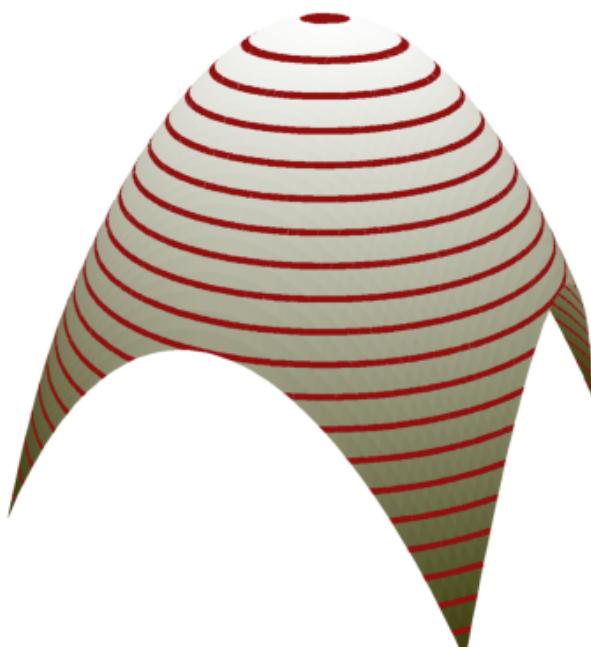
Parameter warna mengubah warna permukaan. Warna garis level juga dapat diubah.

```
>plot3d("-x^2-y^2",color=rgb(0.2,0.2,0),hue=true,frame=false, ...
> zoom=3,contourcolor=red,level=-2:0.1:1,dl=0.01):
```



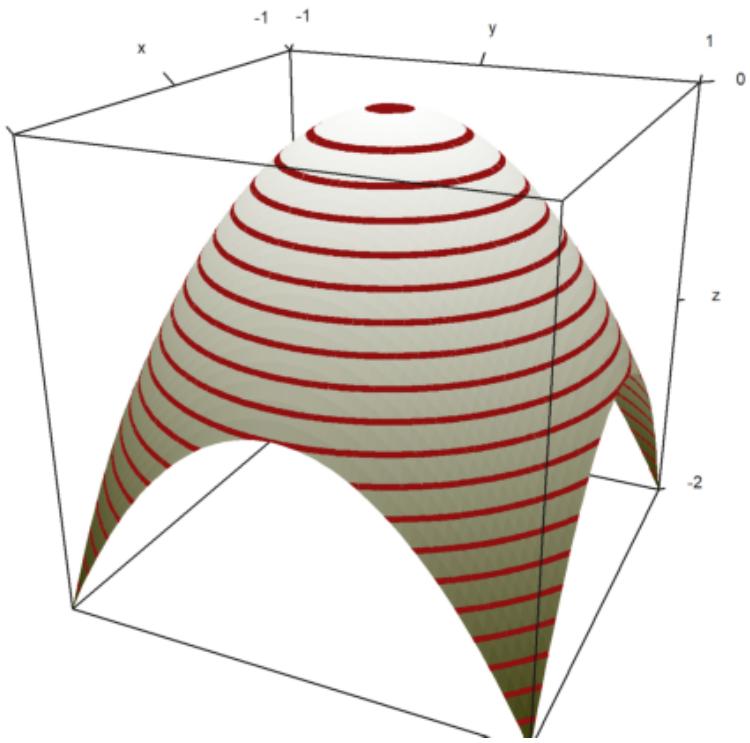
Penggantian perintah `hue=true` menjadi `hue=false` memberikan efek cahaya pada bagian bawah plot. Plot dengan perintah `hue=false` memiliki efek gradasi warna yang cenderung semakin gelap pada bagian bawah plot.

```
>plot3d("-x^2-y^2",color=rgb(0.2,0.2,0),hue=false,frame=false, ...
>   zoom=3,contourcolor=red,level=-2:0.1:1,dl=0.01):
```

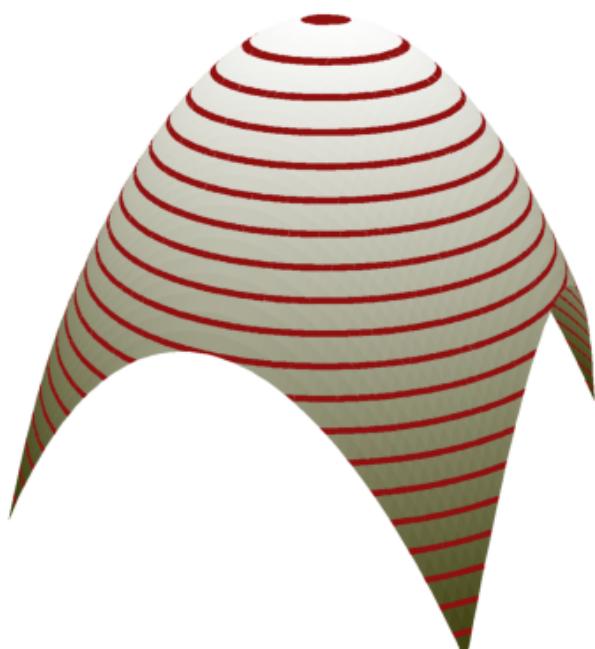


Dibawah ini hasil dari penambahan perintah `frame=true`, sehingga plot memiliki bingkai ketika diprint.

```
>plot3d("-x^2-y^2",color=rgb(0.2,0.2,0),hue=false,frame=true, ...
>   zoom=3,contourcolor=red,level=-2:0.1:1,dl=0.01):
```

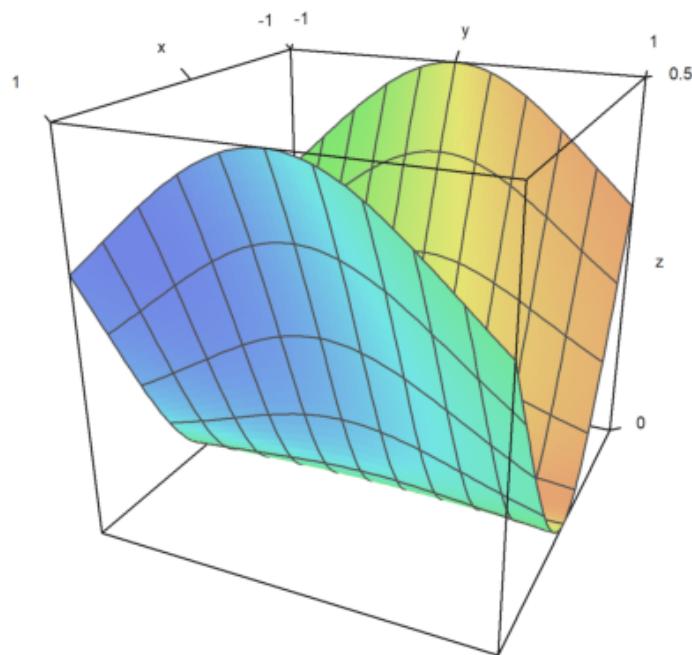


```
>plot3d("-x^2-y^2",color=rgb(0.2,0.2,0),hue=false,frame=false, ...
>   zoom=3,contourcolor=red,level=-2:0.1:1,dl=0.01):
```



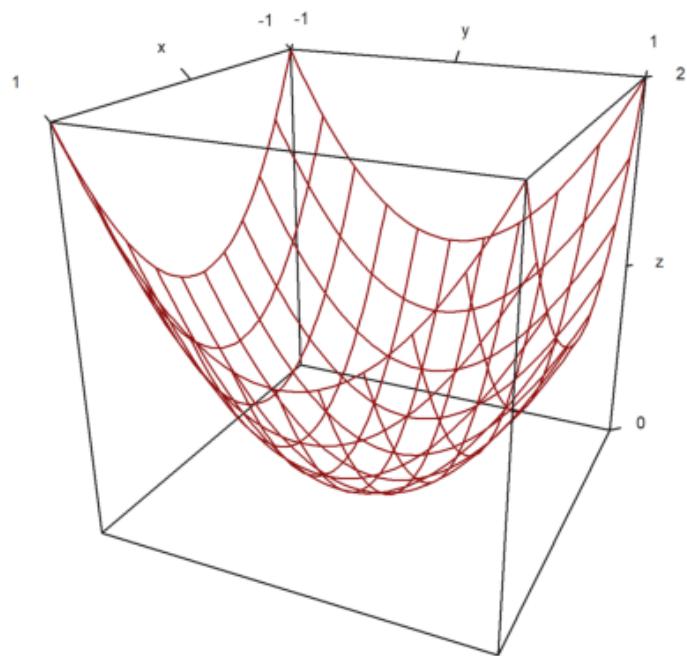
Warna 0 memberikan efek pelangi khusus.

```
>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):
```



Plot Implisit

Ada juga plot implisit dalam tiga dimensi. Euler menghasilkan pemotongan melalui objek. Fitur plot3d termasuk plot implisit. Plot-plot ini menunjukkan himpunan nol dari suatu fungsi dalam tiga variabel. Solusi dari

$$f(x, y, z) = 0$$

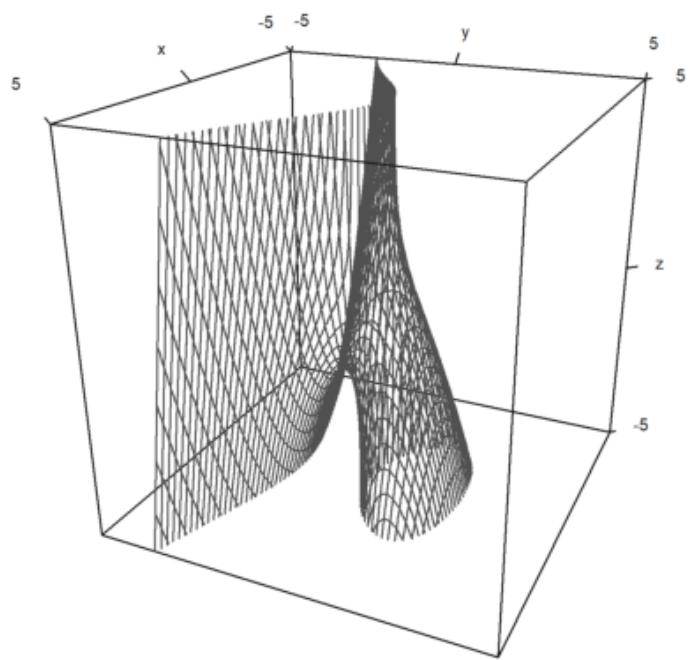
dapat divisualisasikan dalam potongan sejajar dengan bidang x-y-, x-z- dan y-z.

- implicit=1: potong sejajar dengan bidang y-z
- implicit=2: potong sejajar dengan bidang x-z
- implicit=4: potong sejajar dengan bidang x-y

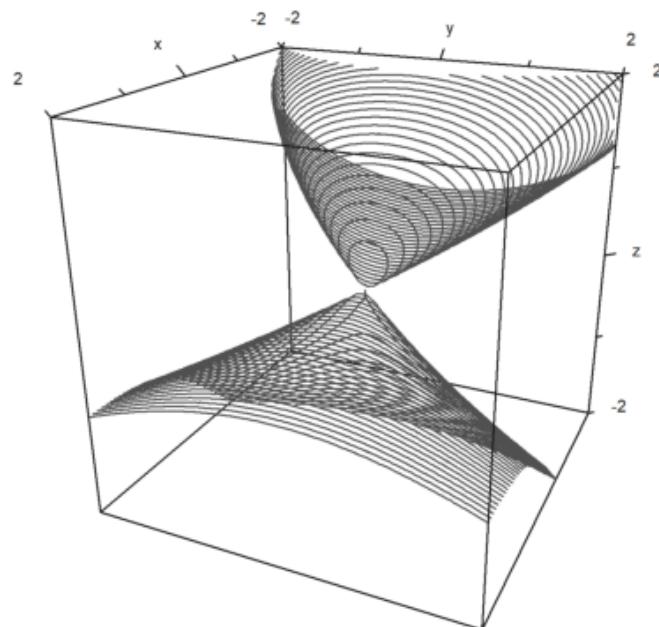
Tambahkan nilai-nilai ini, jika Anda suka. Dalam contoh kita plot

$$M = \{(x, y, z) : x^2 + y^3 + zy = 1\}$$

```
>plot3d("x^2+y^3+z*y-1", r=5, implicit=3):
```



```
>plot3d("x^2+y^2+4*x*z+z^3",>implicit,r=2,zoom=2.5):
```



Merencanakan Data 3D

Sama seperti plot2d, plot3d menerima data. Untuk objek 3D, Anda perlu menyediakan matriks nilai x-, y- dan z, atau tiga fungsi atau ekspresi $f_x(x,y)$, $f_y(x,y)$, $f_z(x,y)$.

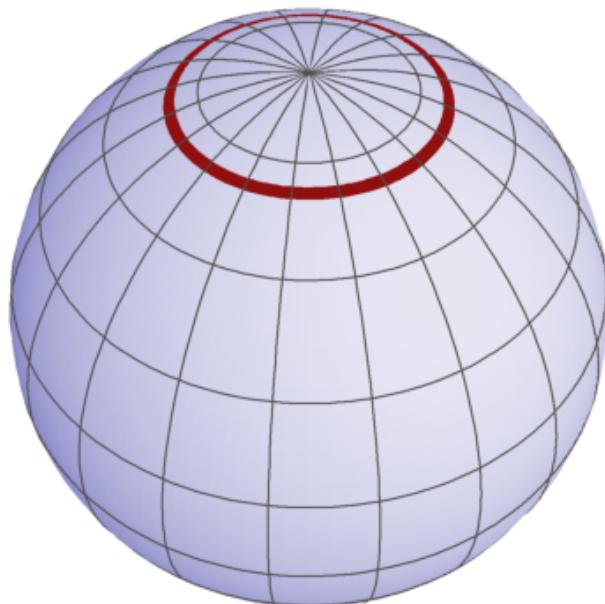
$$\gamma(t, s) = (x(t, s), y(t, s), z(t, s))$$

Karena x,y,z adalah matriks, kita asumsikan bahwa (t,s) melalui sebuah kotak persegi. Hasilnya, Anda dapat memplot gambar persegi panjang di ruang angkasa.

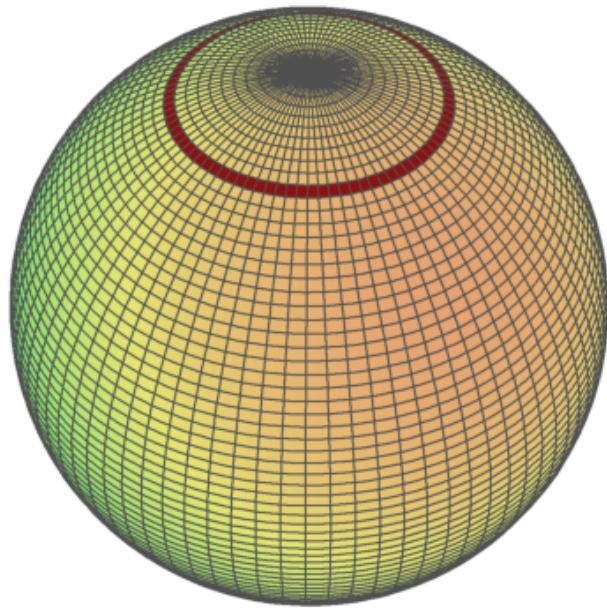
Anda dapat menggunakan bahasa matriks Euler untuk menghasilkan koordinat secara efektif.

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.

```
>t=linspace(0,2pi,180); s=linspace(-pi/2,pi/2,90)'; ...
>x=cos(s)*cos(t); y=cos(s)*sin(t); z=sin(s); ...
>plot3d(x,y,z,>hue, ...
>color=blue,<frame,grid=[10,20], ...
>values=s,contourcolor=red,level=[90°-24°;90°-22°], ...
>scale=1.4,height=50°):
```

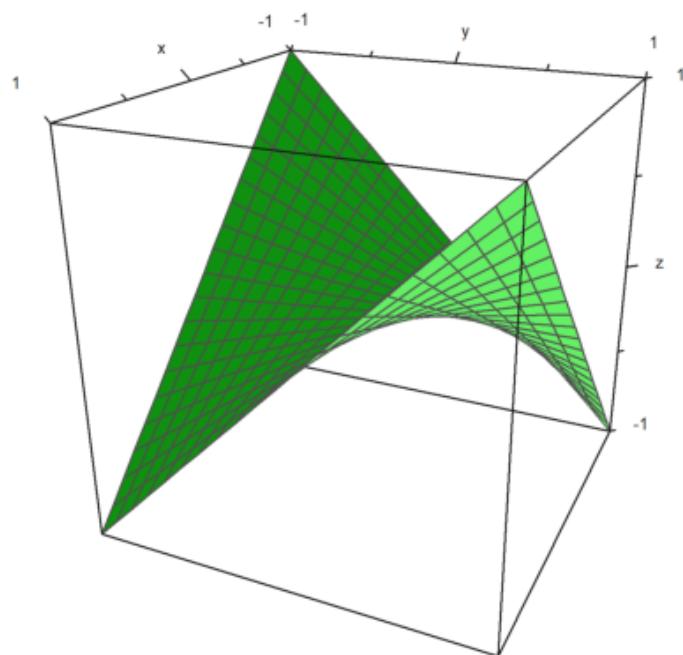


```
>t=linspace(0,2pi,180); s=linspace(-pi/2,pi/2,90)'; ...
>x=cos(s)*cos(t); y=cos(s)*sin(t); z=sin(s); ...
>plot3d(x,y,z,>hue, ...
>color=white,<frame,grid=[100,100], ...
>values=s,contourcolor=red,level=[90°-24°;90°-22°], ...
>scale=1.4,height=50°):
```



Berikut adalah contoh, yang merupakan grafik fungsi.

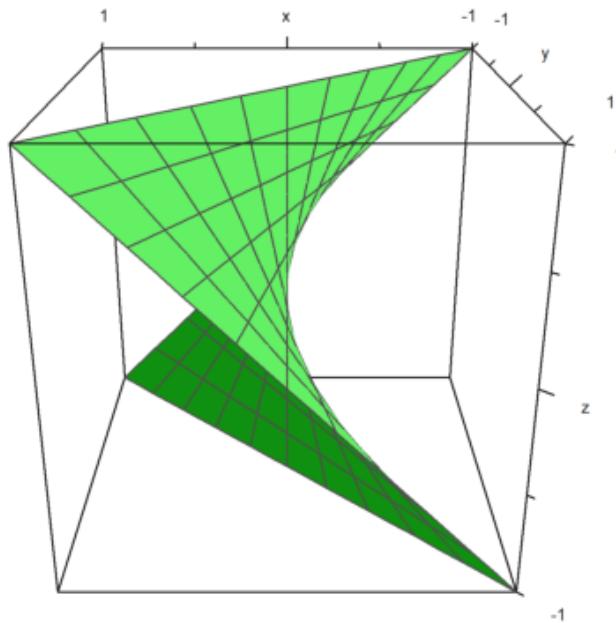
```
>t=-1:0.1:1; s=(-1:0.1:1)'; plot3d(t,s,t*s,grid=110):
```



Namun, kita bisa membuat segala macam permukaan. Berikut adalah permukaan yang sama dengan fungsi.

$$x = y z$$

```
>plot3d(t*s, t, s, angle=180°, grid=10) :
```



Dengan lebih banyak usaha, kami dapat menghasilkan banyak permukaan.

Dalam contoh berikut, kita membuat tampilan bayangan dari bola yang terdistorsi. Koordinat biasa untuk bola adalah

$$\gamma(t, s) = (\cos(t) \cos(s), \sin(t) \sin(s), \cos(s))$$

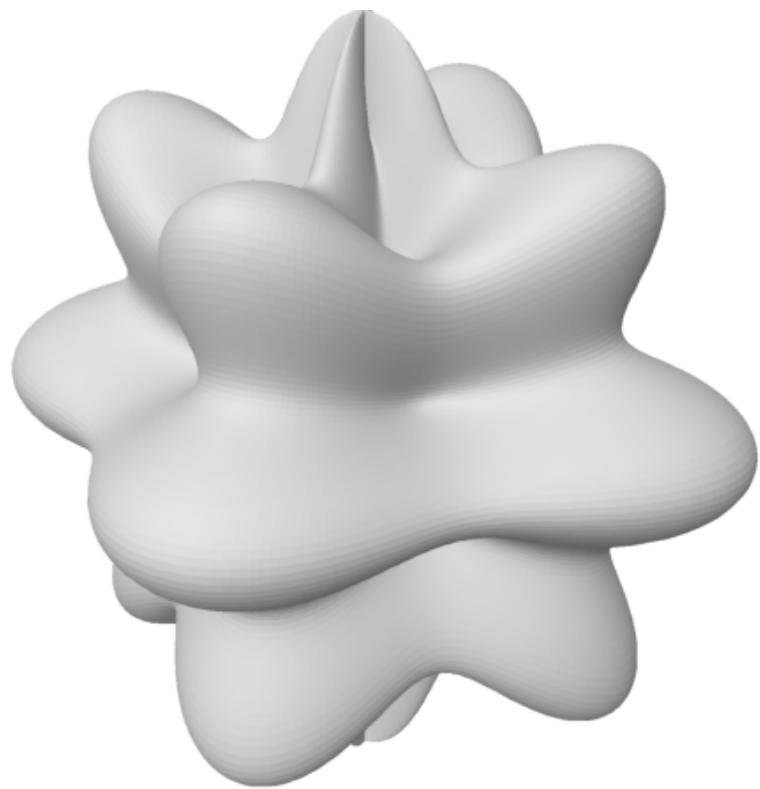
dengan,

$$0 \leq t \leq 2\pi, \quad -\frac{\pi}{2} \leq s \leq \frac{\pi}{2}.$$

Kami mendistorsi ini dengan sebuah faktor

$$d(t, s) = \frac{\cos(4t) + \cos(8s)}{4}.$$

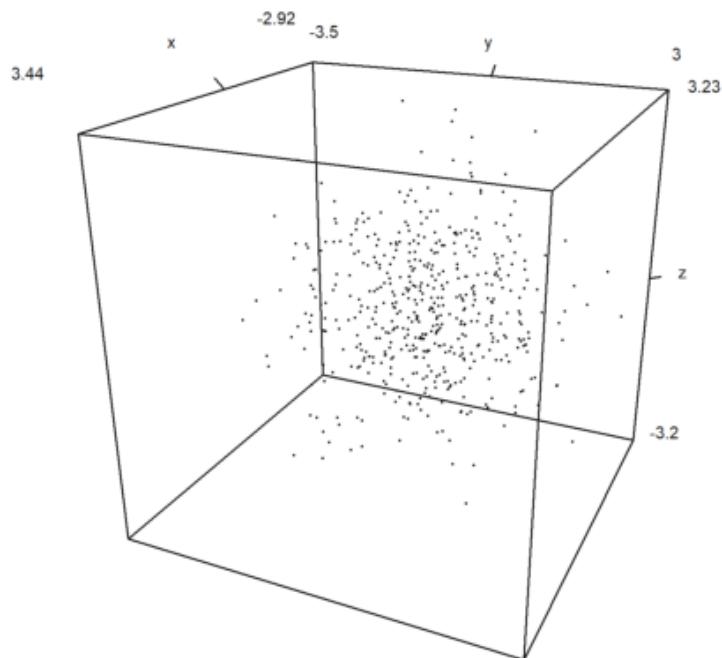
```
>t=linspace(0,2pi,320); s=linspace(-pi/2,pi/2,160)'; ...
>d=1+0.2*(cos(4*t)+cos(8*s)); ...
>plot3d(cos(t)*cos(s)*d, sin(t)*cos(s)*d, sin(s)*d, hue=1, ...
> light=[1,0,1], frame=0, zoom=5) :
```



Tentu saja, titik cloud juga dimungkinkan. Untuk memplot data titik dalam ruang, kita membutuhkan tiga vektor untuk koordinat titik-titik tersebut.

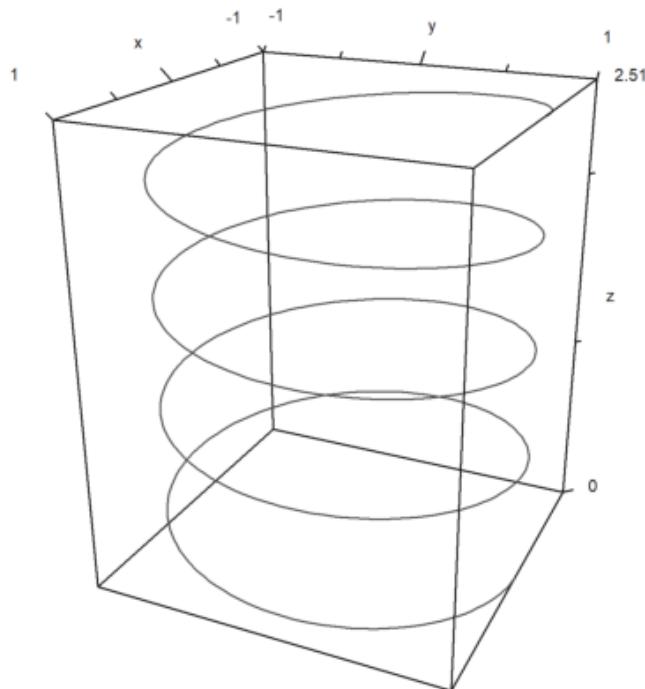
Gayanya sama seperti di plot2d dengan points=true;

```
>n=500; ...  
> plot3d(normal(1,n),normal(1,n),normal(1,n),points=true,style="."):
```

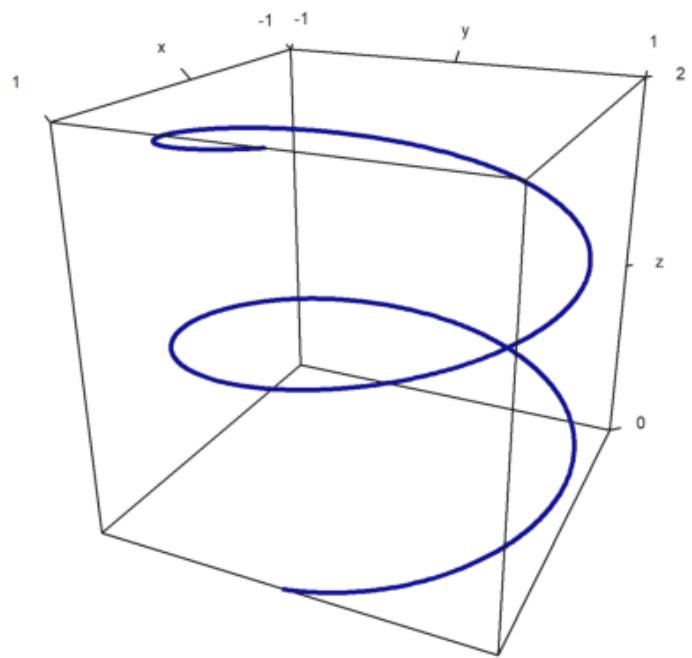


Dimungkinkan juga untuk memplot kurva dalam 3D. Dalam hal ini, lebih mudah untuk menghitung titik-titik kurva. Untuk kurva di pesawat kami menggunakan urutan koordinat dan parameter wire=true.

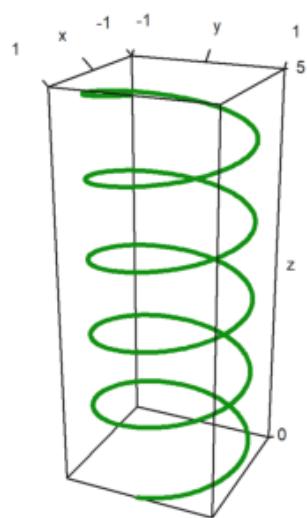
```
>t=linspace(0,8pi,500); ...
>plot3d(sin(t),cos(t),t/10,>wire,zoom=3):
```



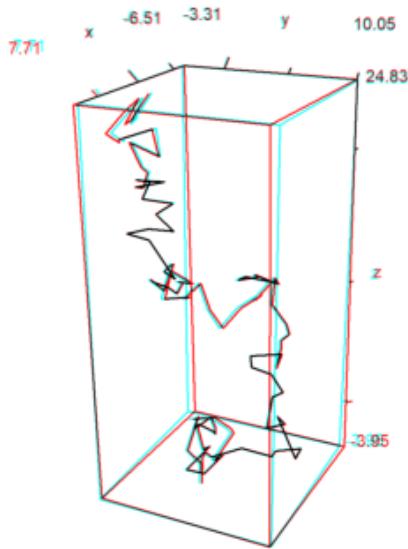
```
>t=linspace(0,4pi,1000); plot3d(cos(t),sin(t),t/2pi,>wire, ...
>linewidth=3,wirecolor=blue):
```



```
>t=linspace(0,10pi,1000); plot3d(cos(t),sin(t),t/2pi,>wire, ...
>linewidth=3, wirecolor=green, disconnect=true):
```

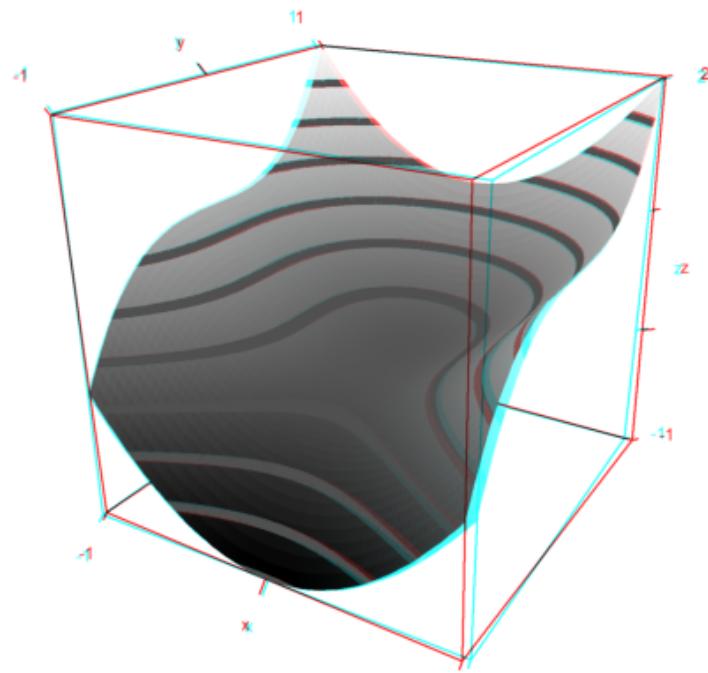


```
>X=cumsum(normal(3,100)); ...
> plot3d(X[1],X[2],X[3],>anaglyph,>wire) :
```



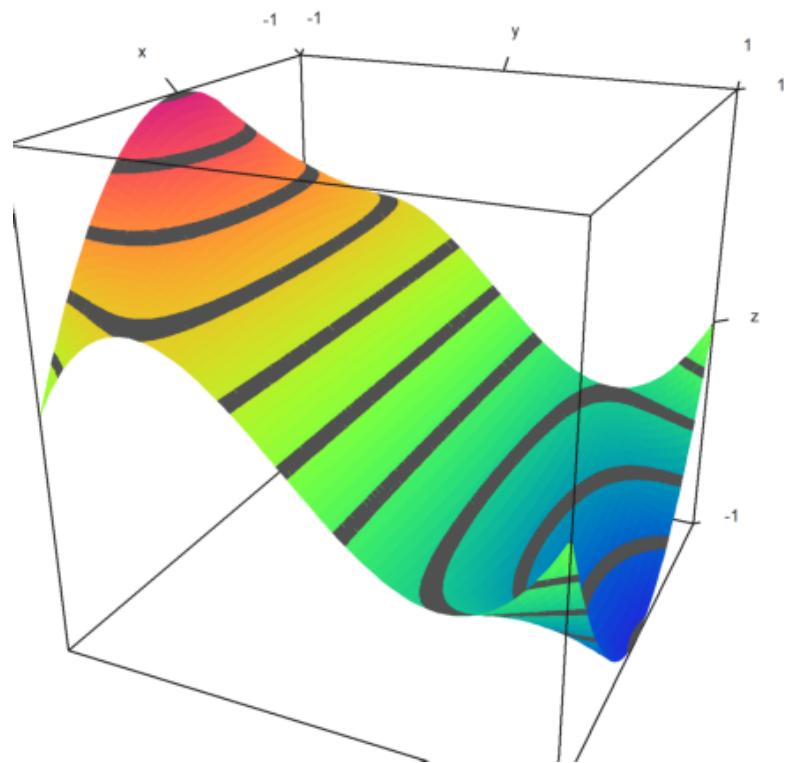
EMT juga dapat memplot dalam mode anaglyph. Untuk melihat plot seperti itu, Anda memerlukan kacamata merah/sian.

```
> plot3d("x^2+y^3",>anaglyph,>contour,angle=30°) :
```



Seringkali, skema warna spektral digunakan untuk plot. Ini menekankan ketinggian fungsi.

```
>plot3d("x^2*y^3-y", >spectral, >contour, zoom=3.2) :
```



Hasil tampilan plot dengan tambahn perintah frame=false dan >anaglyph

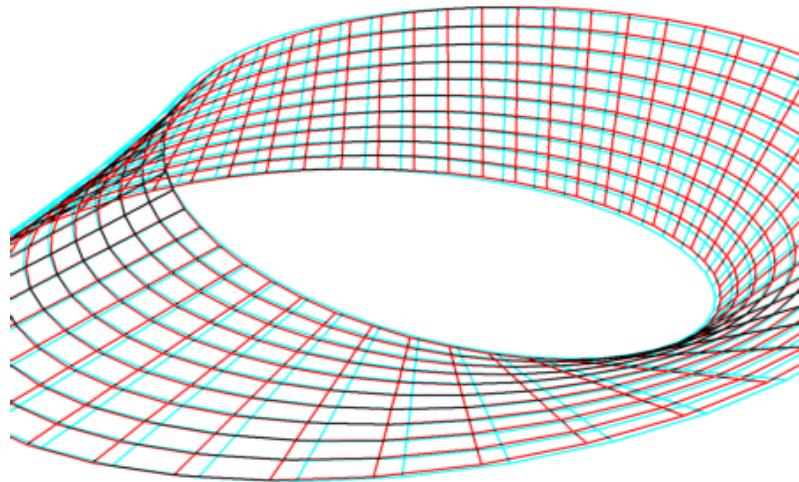
```
>plot3d("x^2*y^3-y",>spectral,>anaglyph,>contour,zoom=3.2,frame=false):
```



Euler juga dapat memplot permukaan berparameter, ketika parameternya adalah nilai x-, y-, dan z dari gambar kotak persegi panjang dalam ruang.

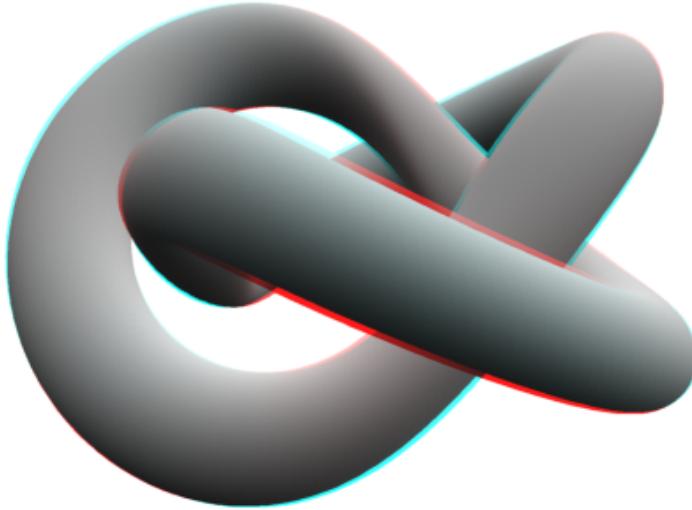
Untuk demo berikut, kami mengatur parameter u- dan v-, dan menghasilkan koordinat ruang dari ini.

```
>u=linspace(-1,1,10); v=linspace(0,2*pi,50)'; ...
>X=(3+u*cos(v/2))*cos(v); Y=(3+u*cos(v/2))*sin(v); Z=u*sin(v/2); ...
>plot3d(X,Y,Z,>anaglyph,<frame,>wire,scale=2.3):
```



Berikut adalah contoh yang lebih rumit, yang megah dengan kacamata merah/sian.

```
>u:=linspace(-pi,pi,160); v:=linspace(-pi,pi,400)'; ...
>x:=(4*(1+.25*sin(3*v))+cos(u))*cos(2*v); ...
>y:=(4*(1+.25*sin(3*v))+cos(u))*sin(2*v); ...
> z=sin(u)+2*cos(3*v); ...
>plot3d(x,y,z,frame=0,scale=1.5,hue=1,light=[1,0,-1],zoom=2.8,>anaglyph):
```



Plot Statistik

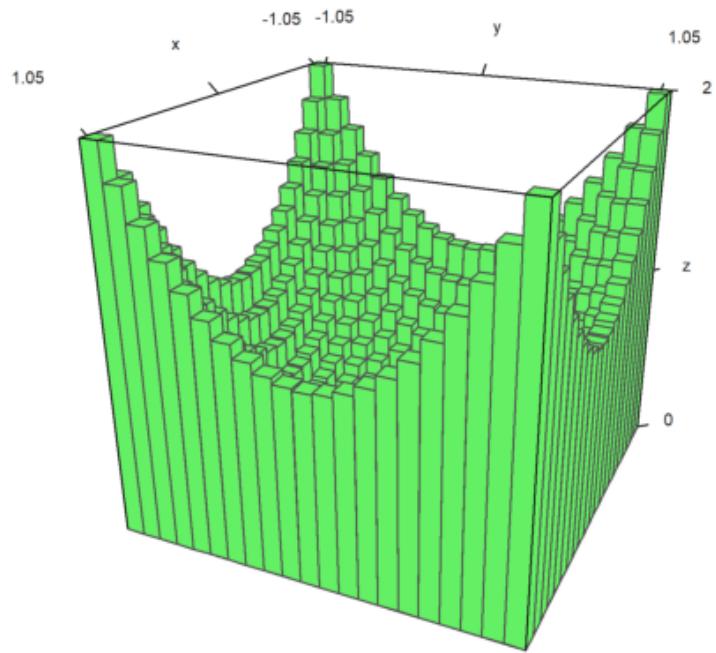
Plot bar juga dimungkinkan. Untuk ini, kita harus menyediakan

- x: vektor baris dengan n+1 elemen
- y: vektor kolom dengan n+1 elemen
- z: matriks nilai nxn.

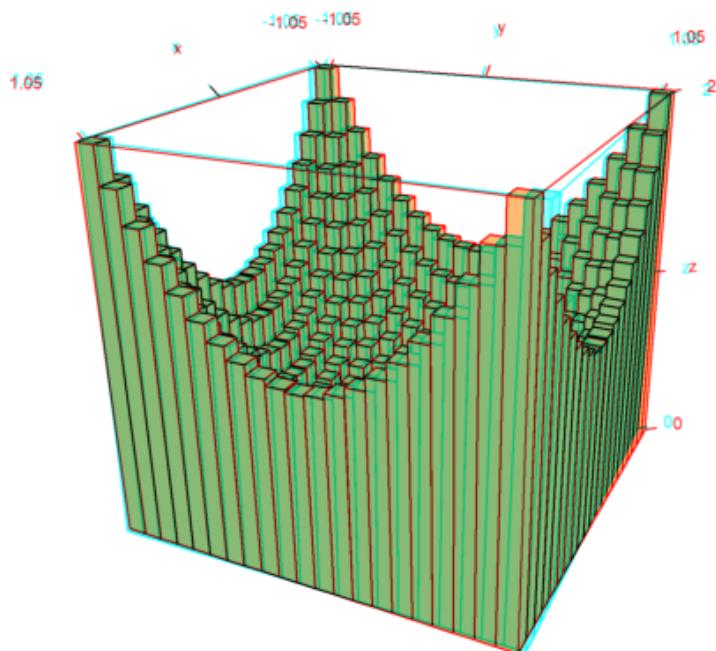
z bisa lebih besar, tetapi hanya nilai nxn yang akan digunakan.

Dalam contoh, pertama-tama kita menghitung nilainya. Kemudian kita sesuaikan x dan y, sehingga vektor 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) :
```

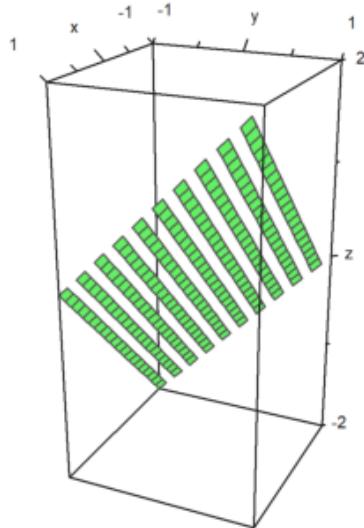


```
>x=-1:0.1:1; y=x'; z=x^2+y^2; ...
>xa=(x|1.1)-0.05; ya=(y_1.1)-0.05; ...
>plot3d(xa,ya,z,bar=true,>anaglyph):
```



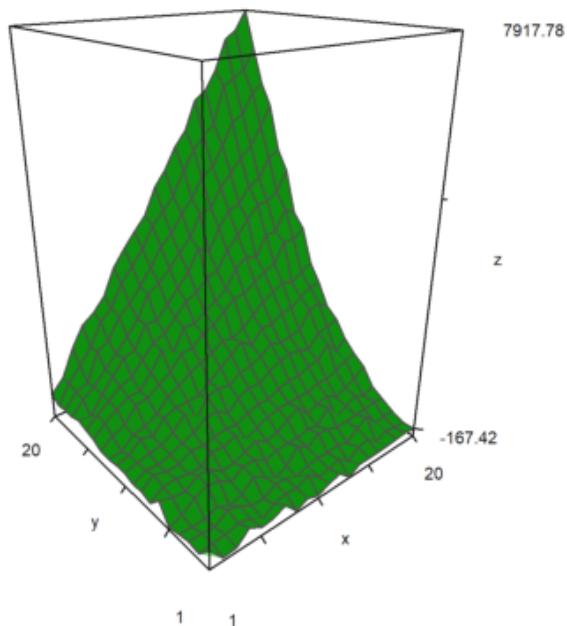
Dimungkinkan untuk membagi plot permukaan menjadi dua atau lebih bagian.

```
>x=-1:0.1:1; y=x'; z=x+y; d=zeros(size(x)); ...
>plot3d(x,y,z,disconnect=2:2:20);
```

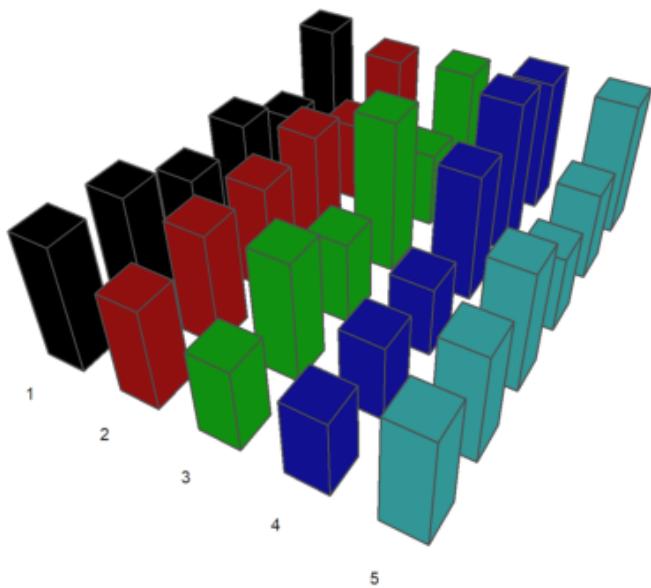


Jika memuat atau menghasilkan matriks data M dari file dan perlu memplotnya dalam 3D, Anda dapat menskalakan matriks ke [-1,1] dengan scale(M), atau menskalakan matriks dengan >zscale. Ini dapat dikombinasikan dengan faktor penskalaan individu yang diterapkan sebagai tambahan.

```
>i=1:20; j=i'; ...
>plot3d(i*j^2+100*normal(20,20),>zscale,scale=[1,1,1.5],angle=-40°,zoom=1.8);
```

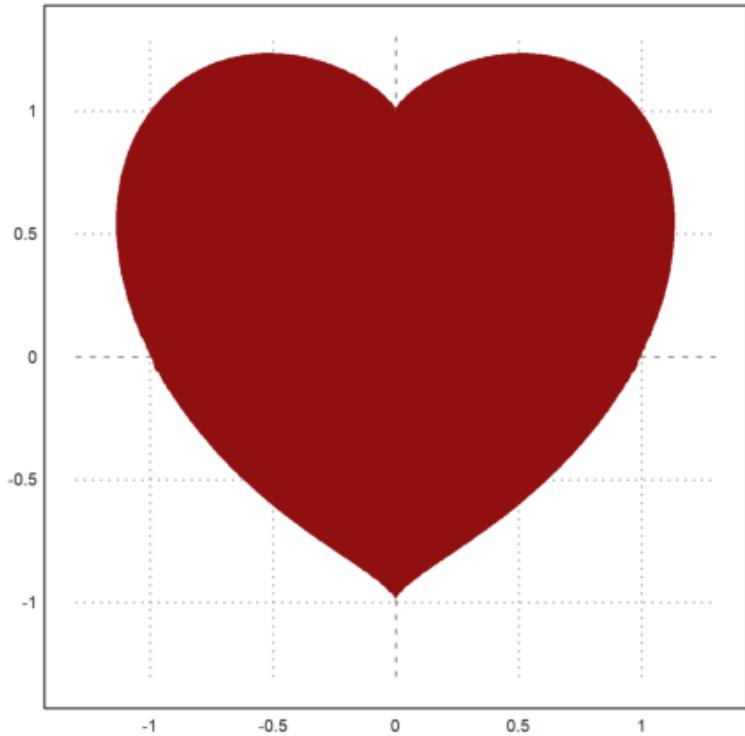


```
>Z=intrandom(5,100,6); v=zeros(5,6); ...
>loop 1 to 5; v[#]=getmultiplicities(1:6,Z[#]); end; ...
>columnsplot3d(v',scols=1:5,ccols=[1:5]):
```



Permukaan Benda Putar

```
>plot2d("(x^2+y^2-1)^3-x^2*y^3",r=1.3, ...
>style="#",color=red,<outline, ...
>level=[-2;0],n=100):
```



```
>ekspressi &= (x^2+y^2-1)^3-x^2*y^3; $ekspressi
```

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

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

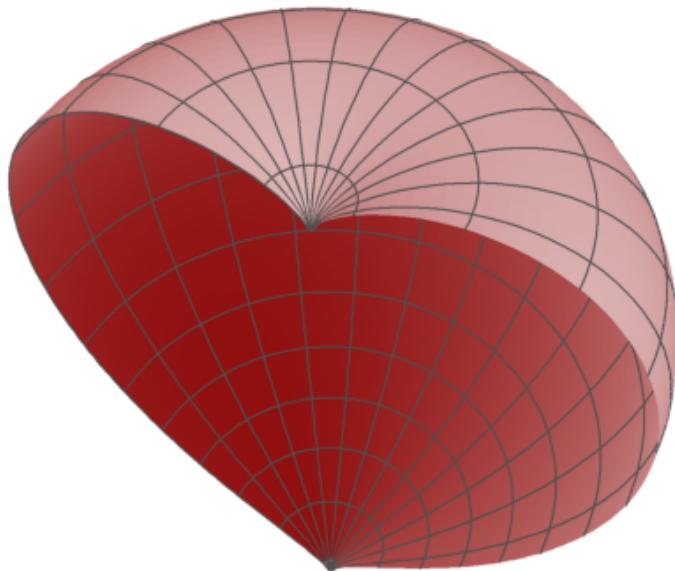
$$x = r \cos(a), \quad y = r \sin(a).$$

```
>function fr(r,a) &= ekspressi with [x=r*cos(a),y=r*sin(a)] | trigreduce; $fr(r,a)
```

$$(r^2 - 1)^3 + \frac{(\sin(5a) - \sin(3a) - 2 \sin a) r^5}{16}$$

Hal ini memungkinkan untuk mendefinisikan fungsi numerik, yang memecahkan r , jika a diberikan. Dengan fungsi itu kita dapat memplot jantung yang diputar sebagai permukaan parametrik.

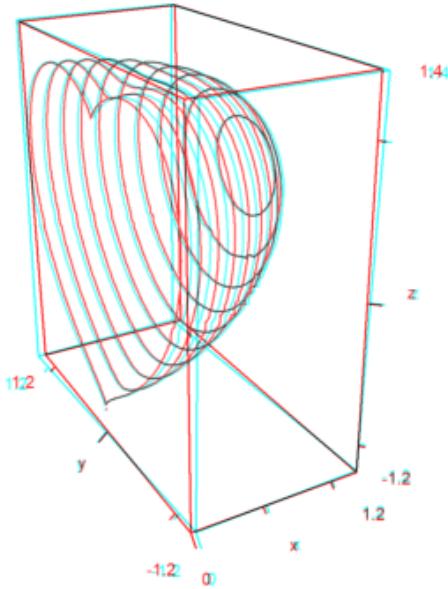
```
>function map f(a) := bisect("fr",0,2;a); ...
>t=linspace(-pi/2,pi/2,100); r=f(t); ...
>s=linspace(pi,2pi,100)'; ...
>plot3d(r*cos(t)*sin(s),r*cos(t)*cos(s),r*sin(t), ...
>>hue,<frame,color=red,zoom=4,amb=0,max=0.7,grid=12,height=50°):
```



Berikut ini adalah plot 3D dari gambar di atas yang diputar di sekitar sumbu z. Kami mendefinisikan fungsi, yang menggambarkan objek.

```
>function f(x,y,z) ...
r=x^2+y^2;
return (r+z^2-1)^3-r*z^3;
endfunction

>plot3d("f(x,y,z)", ...
>xmin=0,xmax=1.2,ymin=-1.2,ymax=1.2,zmin=-1.2,zmax=1.4, ...
>implicit=1,angle=-30°,zoom=2.5,n=[10,60,60],>anaglyph):
```



Plot 3D Khusus

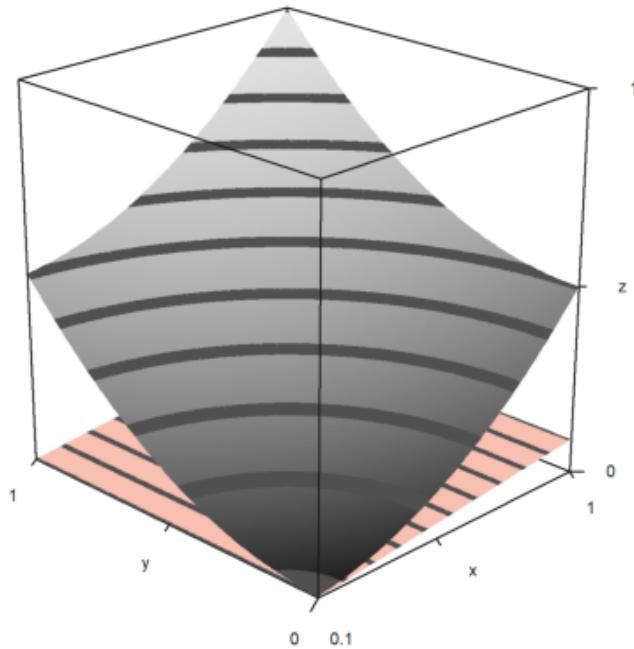
Fungsi `plot3d` bagus untuk dimiliki, tetapi tidak memenuhi semua kebutuhan. Selain rutinitas yang lebih mendasar, dimungkinkan untuk mendapatkan plot berbingkai dari objek apa pun yang Anda suka.

Meskipun Euler bukan program 3D, ia dapat menggabungkan beberapa objek dasar. Kami mencoba memvisualisasikan paraboloid dan garis singgungnya.

```
>function myplot ...
y=0:0.01:1; x=(0.1:0.01:1)';
plot3d(x,y,0.2*(x-0.1)/2,<scale,<frame,>hue, ...
    hues=0.5,>contour,color=orange);
h=holding(1);
plot3d(x,y,(x^2+y^2)/2,<scale,<frame,>contour,>hue);
holding(h);
endfunction
```

Sekarang `framedplot()` menyediakan frame, dan mengatur tampilan.

```
>framedplot("myplot", [0.1,1,0,1,0,1],angle=-45°, ...
> center=[0,0,-0.7],zoom=6):
```

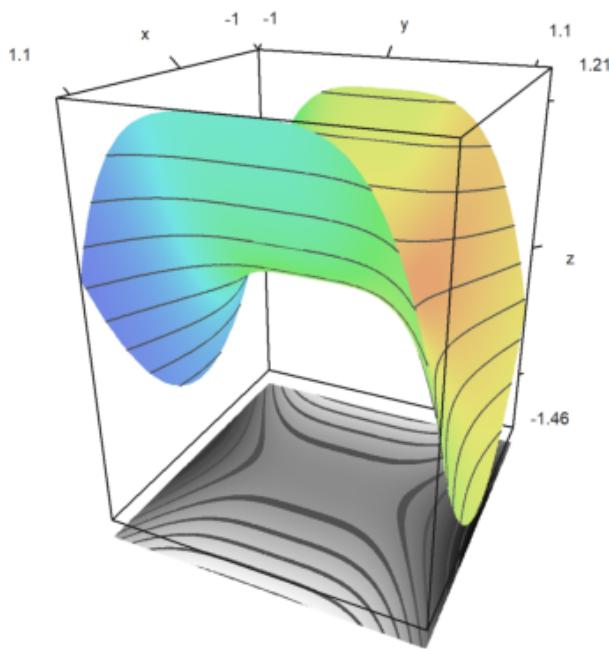


Dengan cara yang sama, Anda dapat memplot bidang kontur secara manual. Perhatikan bahwa `plot3d()` menyetel jendela ke `fullwindow()` secara default, tetapi `plotcontourplane()` mengasumsikan itu.

```
>x=-1:0.02:1.1; y=x'; z=x^2-y^4;
>function myplot (x,y,z) ...
```

```
zoom(2);
wi=fullwindow();
plotcontourplane(x,y,z,level="auto",<scale);
plot3d(x,y,z,>hue,<scale,>add,color=white,level="thin");
window(wi);
reset();
endfunction
```

```
>myplot(x,y,z);
```

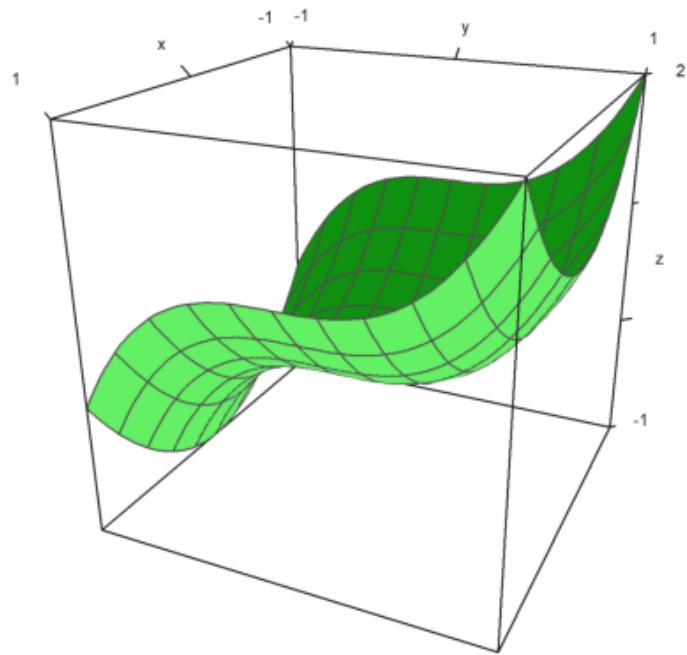


Animasi

Euler dapat menggunakan frame untuk menghitung animasi terlebih dahulu.

Salah satu fungsi yang memanfaatkan teknik ini adalah rotate. Itu dapat mengubah sudut pandang dan menggambar ulang plot 3D. Fungsi memanggil addpage() untuk setiap plot baru. Akhirnya itu menjawai plot. Silakan pelajari sumber rotasi untuk melihat lebih detail.

```
>function testplot () := plot3d("x^2+y^3"); ...
>rotate("testplot"); testplot();
```



Menggambar Povray

Dengan bantuan file Euler povray.e, Euler dapat menghasilkan file Povray. Hasilnya sangat bagus untuk dilihat.

Anda perlu menginstal Povray (32bit atau 64bit) dari <http://www.povray.org/>, dan meletakkan sub-direktori "bin" dari Povray ke jalur lingkungan, atau mengatur variabel "defaultpovray" dengan path lengkap yang menunjuk ke "pvengine.exe".

Antarmuka Povray dari Euler menghasilkan file Povray di direktori home pengguna, dan memanggil Povray untuk mengurai file-file ini. Nama file default adalah current.pov, dan direktori default adalah eulerhome(), biasanya c:\Users\Username\Euler. Povray menghasilkan file PNG, yang dapat dimuat oleh Euler ke dalam buku catatan. Untuk membersihkan file-file ini, gunakan povclear().

Fungsi pov3d memiliki semangat yang sama dengan plot3d. Ini dapat menghasilkan grafik fungsi $f(x,y)$, atau permukaan dengan koordinat X,Y,Z dalam matriks, termasuk garis level opsional. Fungsi ini memulai raytracer secara otomatis, dan memuat adegan ke dalam notebook Euler.

Selain pov3d(), ada banyak fungsi yang menghasilkan objek Povray. Fungsi-fungsi ini mengembalikan string, yang berisi kode Povray untuk objek. Untuk menggunakan fungsi ini, mulai file Povray dengan povstart(). Kemudian gunakan writeln(...) untuk menulis objek ke file adegan. Terakhir, akhiri file dengan povend(). Secara default, raytracer akan dimulai, dan PNG akan dimasukkan ke dalam notebook Euler.

Fungsi objek memiliki parameter yang disebut "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.

Perhatikan bahwa alam semesta Povray memiliki sistem koordinat lain. Antarmuka ini menerjemahkan semua koordinat ke sistem Povray. Jadi Anda dapat terus berpikir dalam sistem koordinat Euler dengan z menunjuk vertikal ke atas, dan x,y,z sumbu dalam arti tangan kanan.

Anda perlu memuat file povray.

```
>load povray;
```

Pastikan, direktori bin Povray ada di jalur. Jika tidak, edit variabel berikut sehingga berisi path ke povray yang dapat dieksekusi.

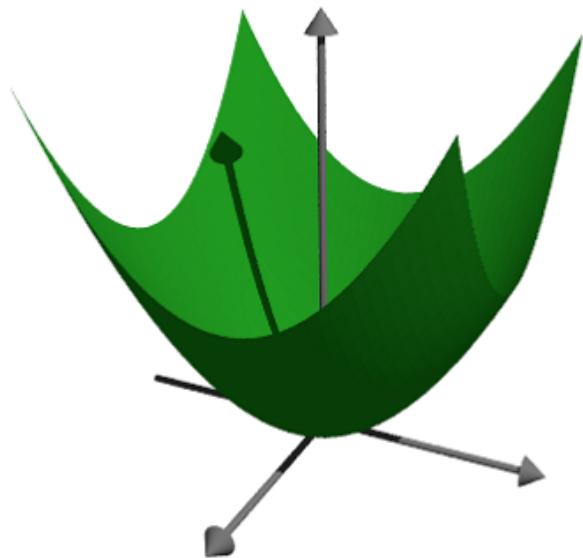
```
>defaultpovray="C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe"
```

C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe

Untuk kesan pertama, kami memplot fungsi sederhana. Perintah berikut menghasilkan file povray di direktori pengguna Anda, dan menjalankan Povray untuk ray tracing file ini.

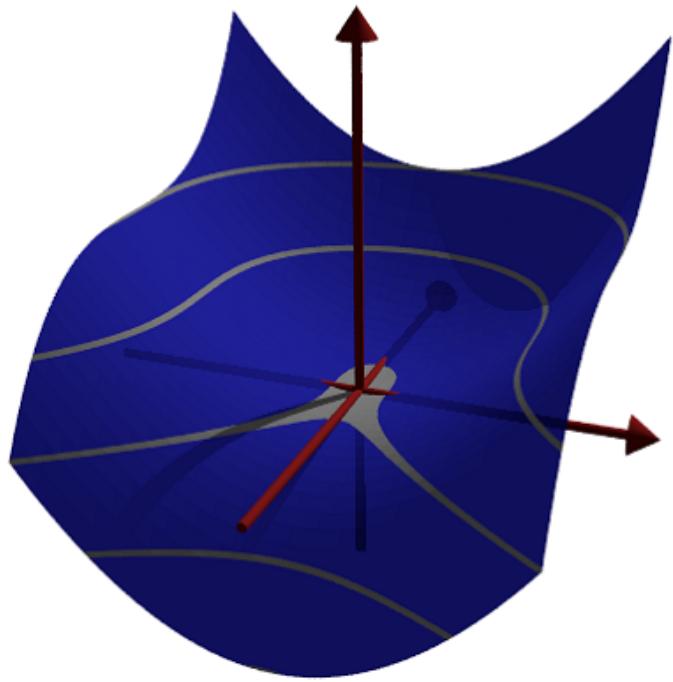
Jika Anda memulai perintah berikut, GUI Povray akan terbuka, menjalankan file, dan menutup secara otomatis. Karena alasan keamanan, Anda akan ditanya, apakah Anda ingin mengizinkan file exe untuk dijalankan. Anda dapat menekan batal untuk menghentikan pertanyaan lebih lanjut. Anda mungkin harus menekan OK di jendela Povray untuk mengakui dialog awal Povray.

```
>pov3d("x^2+y^2", zoom=3);
```



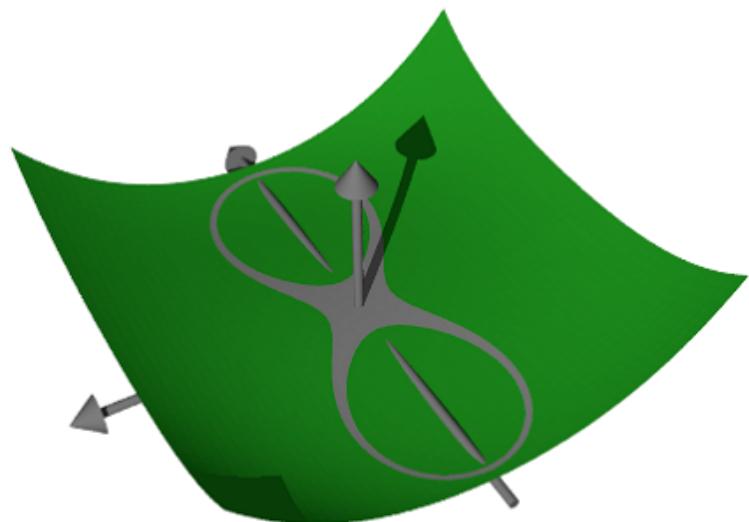
Kita dapat membuat fungsi menjadi transparan dan menambahkan hasil akhir lainnya. Kami juga dapat menambahkan garis level ke plot fungsi.

```
>pov3d("x^2+y^3", axiscolor=red, angle=20°, ...
>   look=povlook(blue, 0.2), level=-1:0.5:1, zoom=3.8);
```



Terkadang perlu untuk mencegah penskalaan fungsi, dan menskalakan fungsi dengan tangan.
Kami memplot himpunan titik di bidang kompleks, di mana produk dari jarak ke 1 dan -1 sama dengan 1.

```
>pov3d("((x-1)^2+y^2)*((x+1)^2+y^2)/40",r=1.5, ...
> angle=-120°,level=1/40,dlevel=0.005,light=[-1,1,1],height=45°,n=50, ...
> <fscale,zoom=3.8);
```



Merencanakan dengan Koordinat

Alih-alih fungsi, kita dapat memplot dengan koordinat. Seperti pada plot3d, kita membutuhkan tiga matriks untuk mendefinisikan objek.

Dalam contoh kita memutar fungsi di sekitar sumbu z.

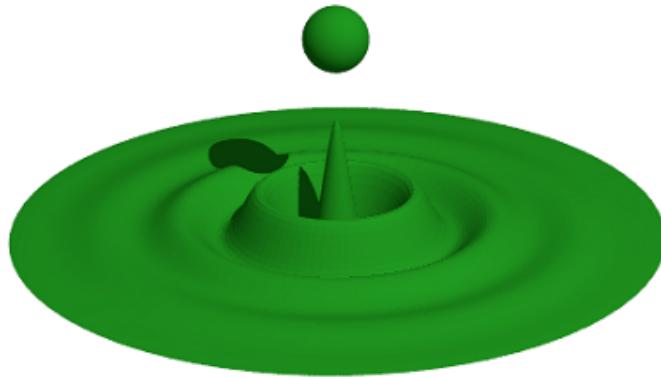
```
>function f(x) := x^3-x+1; ...
>x=-1:0.01:1; t=linspace(0,2pi,8)'; ...
>Z=x; X=cos(t)*f(x); Y=sin(t)*f(x); ...
>pov3d(X,Y,Z,angle=40°,height=20°,axis=0,zoom=4,light=[10,-5,5]);
```



Dalam contoh berikut, kami memplot gelombang teredam. Kami menghasilkan gelombang dengan bahasa matriks Euler.

Kami juga menunjukkan, bagaimana objek tambahan dapat ditambahkan ke adegan pov3d. Untuk pembuatan objek, lihat contoh berikut. Perhatikan bahwa plot3d menskalakan plot, sehingga cocok dengan kubus satuan.

```
>r=linspace(0,1,80); phi=linspace(0,2pi,80)'; ...
>x=r*cos(phi); y=r*sin(phi); z=exp(-5*r)*cos(8*pi*r)/3; ...
>pov3d(x,y,z,zoom=5,axis=0,add=povsphere([0,0,0.5],0.1,povlook(green)), ...
> w=500,h=300);
```



Dengan metode bayangan canggih dari Povray, sangat sedikit titik yang dapat menghasilkan permukaan yang sangat halus. Hanya di perbatasan dan dalam bayang-bayang triknya mungkin menjadi jelas.
Untuk ini, kita perlu menambahkan vektor normal di setiap titik matriks.

```
>Z &= x^2*y^3
```

$$\begin{matrix} 2 & 3 \\ x & y \end{matrix}$$

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);
```

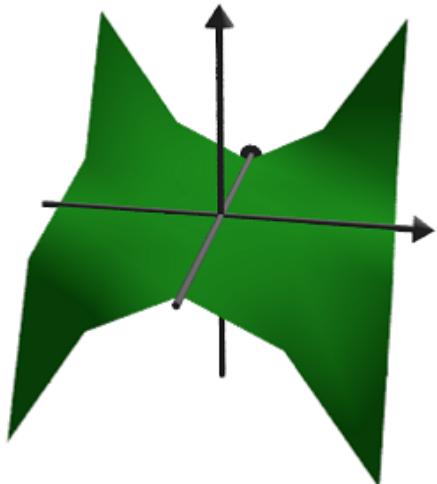
Kami mendefinisikan normal sebagai produk silang dari turunan ini, dan mendefinisikan fungsi koordinat.

```
>N &= crossproduct(dx,dy); NX &= N[1]; NY &= N[2]; NZ &= N[3]; N,
```

$$\begin{bmatrix} 3 & 2 & 2 \\ -2x^2y & -3x^2y & 1 \end{bmatrix}$$

Kita gunakan 25 poin/ titik saja.

```
>x=-1:0.5:1; y=x';
>pov3d(x,y,Z(x,y),angle=10°, ...
> xv=NX(x,y), yv=NY(x,y), zv=NZ(x,y), <shadow>;
```



Berikut ini adalah simpul Trefoil yang dilakukan oleh A. Busser di Povray. Ada versi yang ditingkatkan dari ini dalam contoh.

Lihat: Contoh\Trefoil Simpul | Simpul trefoil

Untuk tampilan yang bagus dengan tidak terlalu banyak titik, kami menambahkan vektor normal di sini. Kami menggunakan Maxima untuk menghitung normal bagi kami. Pertama, ketiga fungsi koordinat sebagai ekspresi simbolik.

```
>X &= ((4+sin(3*y))+cos(x))*cos(2*y); ...
>Y &= ((4+sin(3*y))+cos(x))*sin(2*y); ...
>Z &= sin(x)+2*cos(3*y);
```

Kemudian kedua vektor turunan ke x dan y.

```
>dx &= diff([X,Y,Z],x); dy &= diff([X,Y,Z],y);
```

Sekarang normal, yang merupakan produk silang dari dua turunan.

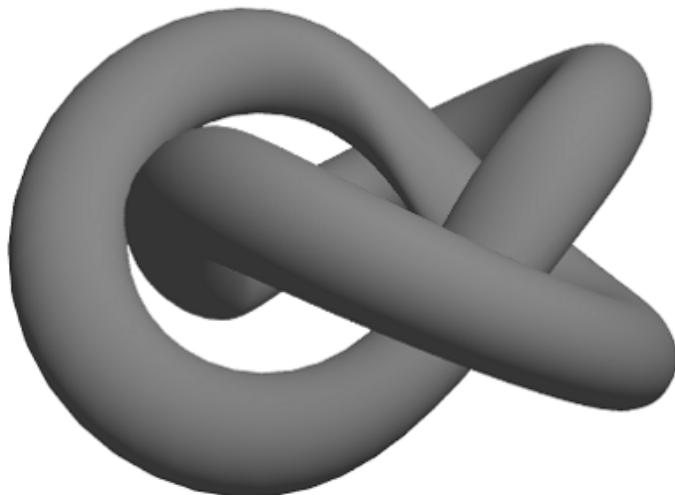
```
>dn &= crossproduct(dx,dy);
```

Kami sekarang mengevaluasi semua ini secara numerik.

```
>x:=linspace(-%pi,%pi,40); y:=linspace(-%pi,%pi,100)';
```

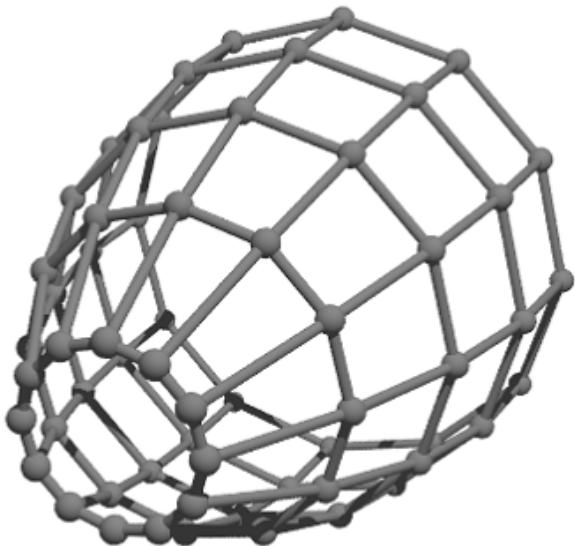
Vektor normal adalah evaluasi dari ekspresi simbolik $dn[i]$ untuk $i=1,2,3$. Sintaks untuk ini adalah &"expression"(parameters). Ini adalah alternatif dari metode pada contoh sebelumnya, di mana kita mendefinisikan ekspresi simbolik NX, NY, NZ terlebih dahulu.

```
>pov3d(X(x,y),Y(x,y),Z(x,y),axis=0,zoom=5,w=450,h=350, ...
>    <shadow,look=povlook(gray), ...
>    xv=&"dn[1] "(x,y), yv=&"dn[2] "(x,y), zv=&"dn[3] "(x,y));
```



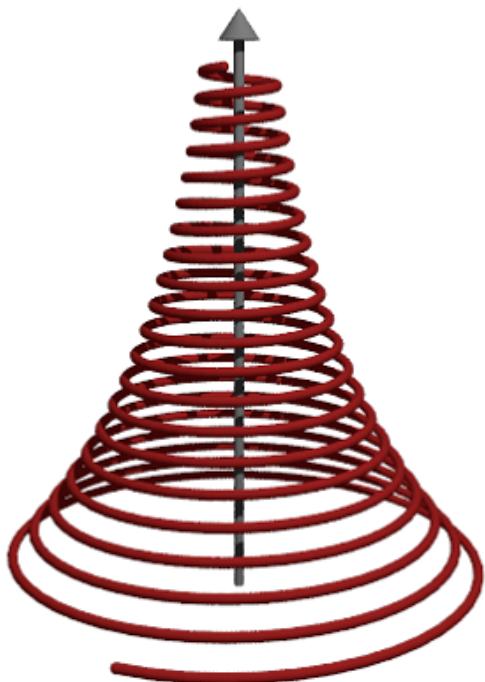
Kami juga dapat menghasilkan grid dalam 3D.

```
>povstart(zoom=4); ...
>x=-1:0.5:1; r=1-(x+1)^2/6; ...
>t=(0°:30°:360°)'; y=r*cos(t); z=r*sin(t); ...
>writeln(povgrid(x,y,z,d=0.02,dballs=0.05)); ...
>povend();
```



Dengan povgrid(), kurva dimungkinkan.

```
>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();
```



Objek Povray

Di atas, kami menggunakan pov3d untuk memplot permukaan. Antarmuka povray di Euler juga dapat menghasilkan objek Povray. Objek-objek ini disimpan sebagai string di Euler, dan perlu ditulis ke file Povray. Kami memulai output dengan povstart().

```
>povstart (zoom=4);
```

Pertama kita mendefinisikan tiga silinder, dan menyimpannya dalam string di Euler. Fungsi povx() dll. hanya mengembalikan vektor [1,0,0], yang dapat digunakan sebagai gantinya.

```
>c1=povcylinder(-povx,povx,1,povlook(red)); ...
>c2=povcylinder(-povy,povy,1,povlook(green)); ...
>c3=povcylinder(-povz,povz,1,povlook(blue)); ...
```

String berisi kode Povray, yang tidak perlu kita pahami pada saat itu.

```
>c1
```

```
cylinder { <-1,0,0>, <1,0,0>, 1
    texture { pigment { color rgb <0.564706,0.0627451,0.0627451> } }
    finish { ambient 0.2 }
}
```

Seperti yang Anda lihat, kami menambahkan tekstur ke objek dalam tiga warna berbeda. Itu dilakukan oleh `povlook()`, yang mengembalikan string dengan kode Povray yang relevan. Kita dapat menggunakan warna Euler default, atau menentukan warna kita sendiri. Kami juga dapat menambahkan transparansi, atau mengubah cahaya sekitar.

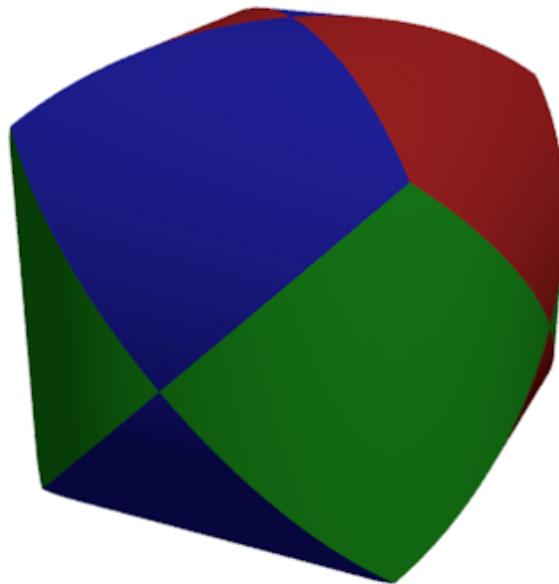
```
>povlook(rgb(0.1,0.2,0.3),0.1,0.5);
```

Sekarang kita mendefinisikan objek persimpangan, dan menulis hasilnya ke file.

```
>writeln(povintersection([c1,c2,c3]));
```

Persimpangan tiga silinder sulit untuk divisualisasikan, jika Anda belum pernah melihatnya sebelumnya.

```
>povend;
```



Fungsi berikut menghasilkan fraktal secara rekursif.

Fungsi pertama menunjukkan, bagaimana Euler menangani objek Povray sederhana. Fungsi `povbox()` mengembalikan string, yang berisi koordinat kotak, tekstur, dan hasil akhir.

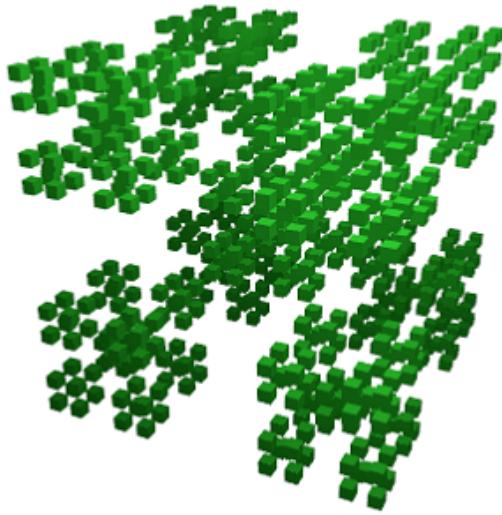
```
>function onebox(x,y,z,d) := povbox([x,y,z],[x+d,y+d,z+d],povlook());  
>function fractal (x,y,z,h,n) ...
```

```
if n==1 then writeln(onebox(x,y,z,h));  
else  
    h=h/3;
```

```
fractal(x,y,z,h,n-1);
fractal(x+2*h,y,z,h,n-1);
fractal(x,y+2*h,z,h,n-1);
fractal(x,y,z+2*h,h,n-1);
fractal(x+2*h,y+2*h,z,h,n-1);
fractal(x+2*h,y,z+2*h,h,n-1);
fractal(x,y+2*h,z+2*h,h,n-1);
fractal(x+2*h,y+2*h,z+2*h,h,n-1);
fractal(x+h,y+h,z+h,h,n-1);
endif;
endfunction
```

```
>povstart(fade=10,<shadow):
```

```
>fractal(-1,-1,-1,2,4);
>povend();
```



Differences allow cutting off one object from another. Like intersections, there are part of the CSG objects of Povray.

```
>povstart (light=[5,-5,5], fade=10);
```

Untuk demonstrasi ini, kami mendefinisikan objek di Povray, alih-alih menggunakan string di Euler. Definisi ditulis ke file segera.

Koordinat kotak -1 berarti [-1,-1,-1].

```
>povdefine ("mycube", povbox (-1,1));
```

Kita dapat menggunakan objek ini di povobject(), yang mengembalikan string seperti biasa.

```
>c1=povobject ("mycube", povlook (red));
```

Kami menghasilkan kubus kedua, dan memutar dan menskalakannya sedikit.

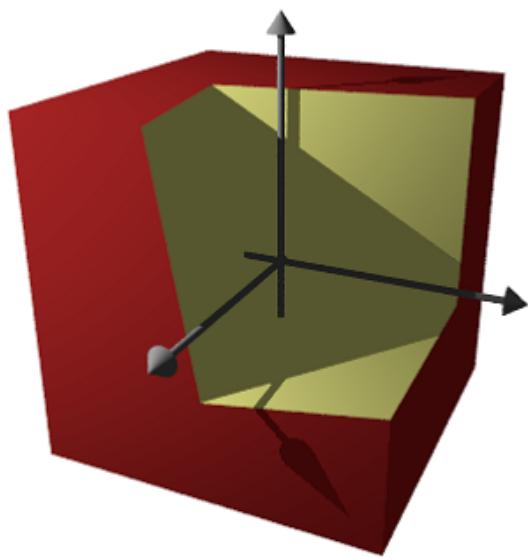
```
>c2=povobject ("mycube", povlook (yellow), translate=[1,1,1], ...
>    rotate=xrotate(10°)+yrotate(10°), scale=1.2);
```

Kemudian kita ambil selisih kedua benda tersebut.

```
>writeln (povdifference (c1,c2));
```

Sekarang tambahkan tiga sumbu.

```
>writeAxis (-1.2,1.2,axis=1); ...
>writeAxis (-1.2,1.2,axis=2); ...
>writeAxis (-1.2,1.2,axis=4); ...
>povend();
```



Fungsi Implisit

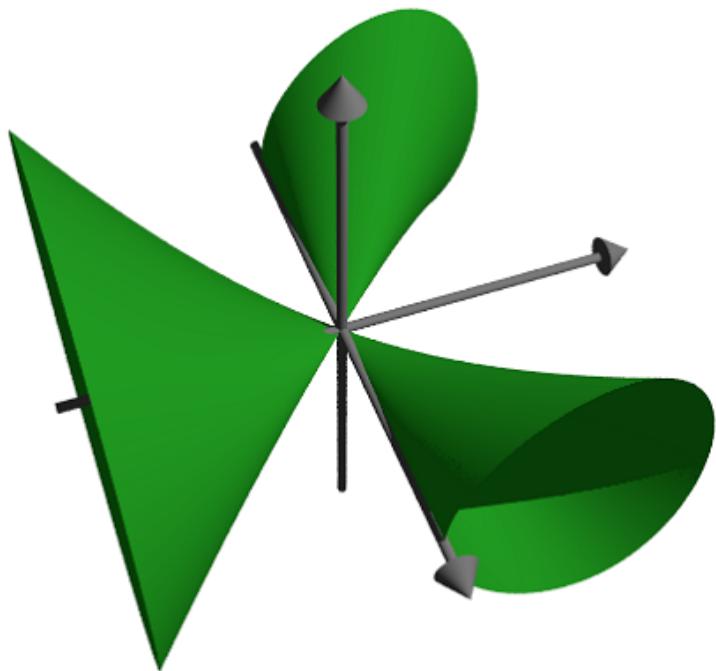
Povray dapat memplot himpunan di mana $f(x,y,z)=0$, seperti parameter implisit di plot3d. Namun, hasilnya terlihat jauh lebih baik.

Sintaks untuk fungsinya sedikit berbeda. Anda tidak dapat menggunakan output dari ekspresi Maxima atau Euler.

```
>povstart (angle=70°,height=50°,zoom=4);
```

Buat permukaan implisit. Perhatikan sintaks yang berbeda dalam ekspresi.

```
>writeln(povsurface ("pow(x,2)*y-pow(y,3)-pow(z,2)",povlook(green))); ...
>writeAxes(); ...
>povend();
```



Objek Jala

Dalam contoh ini, kami menunjukkan cara membuat objek mesh, dan menggambarnya dengan informasi tambahan.

Kami ingin memaksimalkan xy di bawah kondisi $x+y=1$ dan menunjukkan sentuhan tangensial dari garis level.

```
>\povstart (angle=-10°,center=[0.5,0.5,0.5],zoom=7);
```

```
Syntax error in expression, or unfinished expression!
Error in:
\povstart (angle=-10°,center=[0.5,0.5,0.5],zoom=7); ...  
^
```

Kami tidak dapat menyimpan objek dalam string seperti sebelumnya, karena terlalu besar. Jadi kita mendefinisikan objek dalam file Povray menggunakan declare. Fungsi povtriangle() melakukan ini secara otomatis. Itu dapat menerima vektor normal seperti pov3d().

Berikut ini mendefinisikan objek mesh, dan langsung menulisnya ke dalam file.

```
>x=0:0.02:1; y=x'; z=x*y; vx=-y; vy=-x; vz=1;
>mesh=povtriangles(x,y,z,"",vx,vy,vz);
```

```
#declare mesh1 =
mesh2 {
vertex_vectors { 2601,
<0,0,0>,
<0.02,0,0>,
<0.04,0,0>,
<0.06,0,0>,
```

<0.08,0,0>,
<0.1,0,0>,
<0.12,0,0>,
<0.14,0,0>,
<0.16,0,0>,
<0.18,0,0>,
<0.2,0,0>,
<0.22,0,0>,
<0.24,0,0>,
<0.26,0,0>,
<0.28,0,0>,
<0.3,0,0>,
<0.32,0,0>,
<0.34,0,0>,
<0.36,0,0>,
<0.38,0,0>,
<0.4,0,0>,
<0.42,0,0>,
<0.44,0,0>,
<0.46,0,0>,
<0.48,0,0>,
<0.5,0,0>,
<0.52,0,0>,
<0.54,0,0>,
<0.56,0,0>,
<0.58,0,0>,
<0.6,0,0>,
<0.62,0,0>,
<0.64,0,0>,
<0.66,0,0>,
<0.68,0,0>,
<0.7,0,0>,
<0.72,0,0>,
<0.74,0,0>,
<0.76,0,0>,
<0.78,0,0>,
<0.8,0,0>,
<0.82,0,0>,
<0.84,0,0>,
<0.86,0,0>,
<0.88,0,0>,
<0.9,0,0>,
<0.92,0,0>,
<0.94,0,0>,
<0.96,0,0>,
<0.98,0,0>,
<1,0,0>,
<0,0.02,0>,
<0.02,0.02,0.0004>,
<0.04,0.02,0.0008>,
<0.06,0.02,0.0012>,
<0.08,0.02,0.0016>,
<0.1,0.02,0.002>,
<0.12,0.02,0.0024>,
<0.14,0.02,0.0028>,
<0.16,0.02,0.0032>,

<0.18,0.02,0.0036>,
<0.2,0.02,0.004>,
<0.22,0.02,0.0044>,
<0.24,0.02,0.0048>,
<0.26,0.02,0.0052>,
<0.28,0.02,0.0056>,
<0.3,0.02,0.006>,
<0.32,0.02,0.0064>,
<0.34,0.02,0.0068>,
<0.36,0.02,0.0072>,
<0.38,0.02,0.0076>,
<0.4,0.02,0.008>,
<0.42,0.02,0.0084>,
<0.44,0.02,0.0088>,
<0.46,0.02,0.0092>,
<0.48,0.02,0.0096>,
<0.5,0.02,0.01>,
<0.52,0.02,0.0104>,
<0.54,0.02,0.0108>,
<0.56,0.02,0.0112>,
<0.58,0.02,0.0116>,
<0.6,0.02,0.012>,
<0.62,0.02,0.0124>,
<0.64,0.02,0.0128>,
<0.66,0.02,0.0132>,
<0.68,0.02,0.0136>,
<0.7,0.02,0.014>,
<0.72,0.02,0.0144>,
<0.74,0.02,0.0148>,
<0.76,0.02,0.0152>,
<0.78,0.02,0.0156>,
<0.8,0.02,0.016>,
<0.82,0.02,0.0164>,
<0.84,0.02,0.0168>,
<0.86,0.02,0.0172>,
<0.88,0.02,0.0176>,
<0.9,0.02,0.018>,
<0.92,0.02,0.0184>,
<0.94,0.02,0.0188>,
<0.96,0.02,0.0192>,
<0.98,0.02,0.0196>,
<1,0.02,0.02>,
<0,0.04,0>,
<0.02,0.04,0.0008>,
<0.04,0.04,0.0016>,
<0.06,0.04,0.0024>,
<0.08,0.04,0.0032>,
<0.1,0.04,0.004>,
<0.12,0.04,0.0048>,
<0.14,0.04,0.0056>,
<0.16,0.04,0.0064>,
<0.18,0.04,0.0072>,
<0.2,0.04,0.008>,
<0.22,0.04,0.0088>,
<0.24,0.04,0.0096>,
<0.26,0.04,0.0104>,

<0.28,0.04,0.0112>,
<0.3,0.04,0.012>,
<0.32,0.04,0.0128>,
<0.34,0.04,0.0136>,
<0.36,0.04,0.0144>,
<0.38,0.04,0.0152>,
<0.4,0.04,0.016>,
<0.42,0.04,0.0168>,
<0.44,0.04,0.0176>,
<0.46,0.04,0.0184>,
<0.48,0.04,0.0192>,
<0.5,0.04,0.02>,
<0.52,0.04,0.0208>,
<0.54,0.04,0.0216>,
<0.56,0.04,0.0224>,
<0.58,0.04,0.0232>,
<0.6,0.04,0.024>,
<0.62,0.04,0.0248>,
<0.64,0.04,0.0256>,
<0.66,0.04,0.0264>,
<0.68,0.04,0.0272>,
<0.7,0.04,0.028>,
<0.72,0.04,0.0288>,
<0.74,0.04,0.0296>,
<0.76,0.04,0.0304>,
<0.78,0.04,0.0312>,
<0.8,0.04,0.032>,
<0.82,0.04,0.0328>,
<0.84,0.04,0.0336>,
<0.86,0.04,0.0344>,
<0.88,0.04,0.0352>,
<0.9,0.04,0.036>,
<0.92,0.04,0.0368>,
<0.94,0.04,0.0376>,
<0.96,0.04,0.0384>,
<0.98,0.04,0.0392>,
<1,0.04,0.04>,
<0,0.06,0>,
<0.02,0.06,0.0012>,
<0.04,0.06,0.0024>,
<0.06,0.06,0.0036>,
<0.08,0.06,0.0048>,
<0.1,0.06,0.006>,
<0.12,0.06,0.0072>,
<0.14,0.06,0.0084>,
<0.16,0.06,0.0096>,
<0.18,0.06,0.0108>,
<0.2,0.06,0.012>,
<0.22,0.06,0.0132>,
<0.24,0.06,0.0144>,
<0.26,0.06,0.0156>,
<0.28,0.06,0.0168>,
<0.3,0.06,0.018>,
<0.32,0.06,0.0192>,
<0.34,0.06,0.0204>,
<0.36,0.06,0.0216>,

<0.38,0.06,0.0228>,
<0.4,0.06,0.024>,
<0.42,0.06,0.0252>,
<0.44,0.06,0.0264>,
<0.46,0.06,0.0276>,
<0.48,0.06,0.0288>,
<0.5,0.06,0.03>,
<0.52,0.06,0.0312>,
<0.54,0.06,0.0324>,
<0.56,0.06,0.0336>,
<0.58,0.06,0.0348>,
<0.6,0.06,0.036>,
<0.62,0.06,0.0372>,
<0.64,0.06,0.0384>,
<0.66,0.06,0.0396>,
<0.68,0.06,0.0408>,
<0.7,0.06,0.042>,
<0.72,0.06,0.0432>,
<0.74,0.06,0.0444>,
<0.76,0.06,0.0456>,
<0.78,0.06,0.0468>,
<0.8,0.06,0.048>,
<0.82,0.06,0.0492>,
<0.84,0.06,0.0504>,
<0.86,0.06,0.0516>,
<0.88,0.06,0.0528>,
<0.9,0.06,0.054>,
<0.92,0.06,0.0552>,
<0.94,0.06,0.0564>,
<0.96,0.06,0.0576>,
<0.98,0.06,0.0588>,
<1,0.06,0.06>,
<0,0.08,0>,
<0.02,0.08,0.0016>,
<0.04,0.08,0.0032>,
<0.06,0.08,0.0048>,
<0.08,0.08,0.0064>,
<0.1,0.08,0.008>,
<0.12,0.08,0.0096>,
<0.14,0.08,0.0112>,
<0.16,0.08,0.0128>,
<0.18,0.08,0.0144>,
<0.2,0.08,0.016>,
<0.22,0.08,0.0176>,
<0.24,0.08,0.0192>,
<0.26,0.08,0.0208>,
<0.28,0.08,0.0224>,
<0.3,0.08,0.024>,
<0.32,0.08,0.0256>,
<0.34,0.08,0.0272>,
<0.36,0.08,0.0288>,
<0.38,0.08,0.0304>,
<0.4,0.08,0.032>,
<0.42,0.08,0.0336>,
<0.44,0.08,0.0352>,
<0.46,0.08,0.0368>,

<0.48,0.08,0.0384>,
<0.5,0.08,0.04>,
<0.52,0.08,0.0416>,
<0.54,0.08,0.0432>,
<0.56,0.08,0.0448>,
<0.58,0.08,0.0464>,
<0.6,0.08,0.048>,
<0.62,0.08,0.0496>,
<0.64,0.08,0.0512>,
<0.66,0.08,0.0528>,
<0.68,0.08,0.0544>,
<0.7,0.08,0.056>,
<0.72,0.08,0.0576>,
<0.74,0.08,0.0592>,
<0.76,0.08,0.0608>,
<0.78,0.08,0.0624>,
<0.8,0.08,0.064>,
<0.82,0.08,0.0656>,
<0.84,0.08,0.0672>,
<0.86,0.08,0.0688>,
<0.88,0.08,0.0704>,
<0.9,0.08,0.072>,
<0.92,0.08,0.0736>,
<0.94,0.08,0.0752>,
<0.96,0.08,0.0768>,
<0.98,0.08,0.0784>,
<1,0.08,0.08>,
<0,0.1,0>,
<0.02,0.1,0.002>,
<0.04,0.1,0.004>,
<0.06,0.1,0.006>,
<0.08,0.1,0.008>,
<0.1,0.1,0.01>,
<0.12,0.1,0.012>,
<0.14,0.1,0.014>,
<0.16,0.1,0.016>,
<0.18,0.1,0.018>,
<0.2,0.1,0.02>,
<0.22,0.1,0.022>,
<0.24,0.1,0.024>,
<0.26,0.1,0.026>,
<0.28,0.1,0.028>,
<0.3,0.1,0.03>,
<0.32,0.1,0.032>,
<0.34,0.1,0.034>,
<0.36,0.1,0.036>,
<0.38,0.1,0.038>,
<0.4,0.1,0.04>,
<0.42,0.1,0.042>,
<0.44,0.1,0.044>,
<0.46,0.1,0.046>,
<0.48,0.1,0.048>,
<0.5,0.1,0.05>,
<0.52,0.1,0.052>,
<0.54,0.1,0.054>,
<0.56,0.1,0.056>,

<0.58,0.1,0.058>,
<0.6,0.1,0.06>,
<0.62,0.1,0.062>,
<0.64,0.1,0.064>,
<0.66,0.1,0.066>,
<0.68,0.1,0.068>,
<0.7,0.1,0.07>,
<0.72,0.1,0.072>,
<0.74,0.1,0.074>,
<0.76,0.1,0.076>,
<0.78,0.1,0.078>,
<0.8,0.1,0.08>,
<0.82,0.1,0.082>,
<0.84,0.1,0.084>,
<0.86,0.1,0.086>,
<0.88,0.1,0.088>,
<0.9,0.1,0.09>,
<0.92,0.1,0.092>,
<0.94,0.1,0.094>,
<0.96,0.1,0.096>,
<0.98,0.1,0.098>,
<1,0.1,0.1>,
<0,0.12,0>,
<0.02,0.12,0.0024>,
<0.04,0.12,0.0048>,
<0.06,0.12,0.0072>,
<0.08,0.12,0.0096>,
<0.1,0.12,0.012>,
<0.12,0.12,0.0144>,
<0.14,0.12,0.0168>,
<0.16,0.12,0.0192>,
<0.18,0.12,0.0216>,
<0.2,0.12,0.024>,
<0.22,0.12,0.0264>,
<0.24,0.12,0.0288>,
<0.26,0.12,0.0312>,
<0.28,0.12,0.0336>,
<0.3,0.12,0.036>,
<0.32,0.12,0.0384>,
<0.34,0.12,0.0408>,
<0.36,0.12,0.0432>,
<0.38,0.12,0.0456>,
<0.4,0.12,0.048>,
<0.42,0.12,0.0504>,
<0.44,0.12,0.0528>,
<0.46,0.12,0.0552>,
<0.48,0.12,0.0576>,
<0.5,0.12,0.06>,
<0.52,0.12,0.0624>,
<0.54,0.12,0.0648>,
<0.56,0.12,0.0672>,
<0.58,0.12,0.0696>,
<0.6,0.12,0.072>,
<0.62,0.12,0.0744>,
<0.64,0.12,0.0768>,
<0.66,0.12,0.0792>,

<0.68,0.12,0.0816>,
<0.7,0.12,0.084>,
<0.72,0.12,0.0864>,
<0.74,0.12,0.0888>,
<0.76,0.12,0.0912>,
<0.78,0.12,0.0936>,
<0.8,0.12,0.096>,
<0.82,0.12,0.0984>,
<0.84,0.12,0.1008>,
<0.86,0.12,0.1032>,
<0.88,0.12,0.1056>,
<0.9,0.12,0.108>,
<0.92,0.12,0.1104>,
<0.94,0.12,0.1128>,
<0.96,0.12,0.1152>,
<0.98,0.12,0.1176>,
<1,0.12,0.12>,
<0,0.14,0>,
<0.02,0.14,0.0028>,
<0.04,0.14,0.0056>,
<0.06,0.14,0.0084>,
<0.08,0.14,0.0112>,
<0.1,0.14,0.014>,
<0.12,0.14,0.0168>,
<0.14,0.14,0.0196>,
<0.16,0.14,0.0224>,
<0.18,0.14,0.0252>,
<0.2,0.14,0.028>,
<0.22,0.14,0.0308>,
<0.24,0.14,0.0336>,
<0.26,0.14,0.0364>,
<0.28,0.14,0.0392>,
<0.3,0.14,0.042>,
<0.32,0.14,0.0448>,
<0.34,0.14,0.0476>,
<0.36,0.14,0.0504>,
<0.38,0.14,0.0532>,
<0.4,0.14,0.056>,
<0.42,0.14,0.0588>,
<0.44,0.14,0.0616>,
<0.46,0.14,0.0644>,
<0.48,0.14,0.0672>,
<0.5,0.14,0.07>,
<0.52,0.14,0.0728>,
<0.54,0.14,0.0756>,
<0.56,0.14,0.0784>,
<0.58,0.14,0.0812>,
<0.6,0.14,0.084>,
<0.62,0.14,0.0868>,
<0.64,0.14,0.0896>,
<0.66,0.14,0.0924>,
<0.68,0.14,0.0952>,
<0.7,0.14,0.098>,
<0.72,0.14,0.1008>,
<0.74,0.14,0.1036>,
<0.76,0.14,0.1064>,

<0.78,0.14,0.1092>,
<0.8,0.14,0.112>,
<0.82,0.14,0.1148>,
<0.84,0.14,0.1176>,
<0.86,0.14,0.1204>,
<0.88,0.14,0.1232>,
<0.9,0.14,0.126>,
<0.92,0.14,0.1288>,
<0.94,0.14,0.1316>,
<0.96,0.14,0.1344>,
<0.98,0.14,0.1372>,
<1,0.14,0.14>,
<0,0.16,0>,
<0.02,0.16,0.0032>,
<0.04,0.16,0.0064>,
<0.06,0.16,0.0096>,
<0.08,0.16,0.0128>,
<0.1,0.16,0.016>,
<0.12,0.16,0.0192>,
<0.14,0.16,0.0224>,
<0.16,0.16,0.0256>,
<0.18,0.16,0.0288>,
<0.2,0.16,0.032>,
<0.22,0.16,0.0352>,
<0.24,0.16,0.0384>,
<0.26,0.16,0.0416>,
<0.28,0.16,0.0448>,
<0.3,0.16,0.048>,
<0.32,0.16,0.0512>,
<0.34,0.16,0.0544>,
<0.36,0.16,0.0576>,
<0.38,0.16,0.0608>,
<0.4,0.16,0.064>,
<0.42,0.16,0.0672>,
<0.44,0.16,0.0704>,
<0.46,0.16,0.0736>,
<0.48,0.16,0.0768>,
<0.5,0.16,0.08>,
<0.52,0.16,0.0832>,
<0.54,0.16,0.0864>,
<0.56,0.16,0.0896>,
<0.58,0.16,0.0928>,
<0.6,0.16,0.096>,
<0.62,0.16,0.0992>,
<0.64,0.16,0.1024>,
<0.66,0.16,0.1056>,
<0.68,0.16,0.1088>,
<0.7,0.16,0.112>,
<0.72,0.16,0.1152>,
<0.74,0.16,0.1184>,
<0.76,0.16,0.1216>,
<0.78,0.16,0.1248>,
<0.8,0.16,0.128>,
<0.82,0.16,0.1312>,
<0.84,0.16,0.1344>,
<0.86,0.16,0.1376>,

<0.88,0.16,0.1408>,
<0.9,0.16,0.144>,
<0.92,0.16,0.1472>,
<0.94,0.16,0.1504>,
<0.96,0.16,0.1536>,
<0.98,0.16,0.1568>,
<1,0.16,0.16>,
<0,0.18,0>,
<0.02,0.18,0.0036>,
<0.04,0.18,0.0072>,
<0.06,0.18,0.0108>,
<0.08,0.18,0.0144>,
<0.1,0.18,0.018>,
<0.12,0.18,0.0216>,
<0.14,0.18,0.0252>,
<0.16,0.18,0.0288>,
<0.18,0.18,0.0324>,
<0.2,0.18,0.036>,
<0.22,0.18,0.0396>,
<0.24,0.18,0.0432>,
<0.26,0.18,0.0468>,
<0.28,0.18,0.0504>,
<0.3,0.18,0.054>,
<0.32,0.18,0.0576>,
<0.34,0.18,0.0612>,
<0.36,0.18,0.0648>,
<0.38,0.18,0.0684>,
<0.4,0.18,0.072>,
<0.42,0.18,0.0756>,
<0.44,0.18,0.0792>,
<0.46,0.18,0.0828>,
<0.48,0.18,0.0864>,
<0.5,0.18,0.09>,
<0.52,0.18,0.0936>,
<0.54,0.18,0.0972>,
<0.56,0.18,0.1008>,
<0.58,0.18,0.1044>,
<0.6,0.18,0.108>,
<0.62,0.18,0.1116>,
<0.64,0.18,0.1152>,
<0.66,0.18,0.1188>,
<0.68,0.18,0.1224>,
<0.7,0.18,0.126>,
<0.72,0.18,0.1296>,
<0.74,0.18,0.1332>,
<0.76,0.18,0.1368>,
<0.78,0.18,0.1404>,
<0.8,0.18,0.144>,
<0.82,0.18,0.1476>,
<0.84,0.18,0.1512>,
<0.86,0.18,0.1548>,
<0.88,0.18,0.1584>,
<0.9,0.18,0.162>,
<0.92,0.18,0.1656>,
<0.94,0.18,0.1692>,
<0.96,0.18,0.1728>,

<0.98,0.18,0.1764>,
<1,0.18,0.18>,
<0,0.2,0>,
<0.02,0.2,0.004>,
<0.04,0.2,0.008>,
<0.06,0.2,0.012>,
<0.08,0.2,0.016>,
<0.1,0.2,0.02>,
<0.12,0.2,0.024>,
<0.14,0.2,0.028>,
<0.16,0.2,0.032>,
<0.18,0.2,0.036>,
<0.2,0.2,0.04>,
<0.22,0.2,0.044>,
<0.24,0.2,0.048>,
<0.26,0.2,0.052>,
<0.28,0.2,0.056>,
<0.3,0.2,0.06>,
<0.32,0.2,0.064>,
<0.34,0.2,0.068>,
<0.36,0.2,0.072>,
<0.38,0.2,0.076>,
<0.4,0.2,0.08>,
<0.42,0.2,0.084>,
<0.44,0.2,0.088>,
<0.46,0.2,0.092>,
<0.48,0.2,0.096>,
<0.5,0.2,0.1>,
<0.52,0.2,0.104>,
<0.54,0.2,0.108>,
<0.56,0.2,0.112>,
<0.58,0.2,0.116>,
<0.6,0.2,0.12>,
<0.62,0.2,0.124>,
<0.64,0.2,0.128>,
<0.66,0.2,0.132>,
<0.68,0.2,0.136>,
<0.7,0.2,0.14>,
<0.72,0.2,0.144>,
<0.74,0.2,0.148>,
<0.76,0.2,0.152>,
<0.78,0.2,0.156>,
<0.8,0.2,0.16>,
<0.82,0.2,0.164>,
<0.84,0.2,0.168>,
<0.86,0.2,0.172>,
<0.88,0.2,0.176>,
<0.9,0.2,0.18>,
<0.92,0.2,0.184>,
<0.94,0.2,0.188>,
<0.96,0.2,0.192>,
<0.98,0.2,0.196>,
<1,0.2,0.2>,
<0,0.22,0>,
<0.02,0.22,0.0044>,
<0.04,0.22,0.0088>,

<0.06,0.22,0.0132>,
<0.08,0.22,0.0176>,
<0.1,0.22,0.022>,
<0.12,0.22,0.0264>,
<0.14,0.22,0.0308>,
<0.16,0.22,0.0352>,
<0.18,0.22,0.0396>,
<0.2,0.22,0.044>,
<0.22,0.22,0.0484>,
<0.24,0.22,0.0528>,
<0.26,0.22,0.0572>,
<0.28,0.22,0.0616>,
<0.3,0.22,0.066>,
<0.32,0.22,0.0704>,
<0.34,0.22,0.0748>,
<0.36,0.22,0.0792>,
<0.38,0.22,0.0836>,
<0.4,0.22,0.088>,
<0.42,0.22,0.0924>,
<0.44,0.22,0.0968>,
<0.46,0.22,0.1012>,
<0.48,0.22,0.1056>,
<0.5,0.22,0.11>,
<0.52,0.22,0.1144>,
<0.54,0.22,0.1188>,
<0.56,0.22,0.1232>,
<0.58,0.22,0.1276>,
<0.6,0.22,0.132>,
<0.62,0.22,0.1364>,
<0.64,0.22,0.1408>,
<0.66,0.22,0.1452>,
<0.68,0.22,0.1496>,
<0.7,0.22,0.154>,
<0.72,0.22,0.1584>,
<0.74,0.22,0.1628>,
<0.76,0.22,0.1672>,
<0.78,0.22,0.1716>,
<0.8,0.22,0.176>,
<0.82,0.22,0.1804>,
<0.84,0.22,0.1848>,
<0.86,0.22,0.1892>,
<0.88,0.22,0.1936>,
<0.9,0.22,0.198>,
<0.92,0.22,0.2024>,
<0.94,0.22,0.2068>,
<0.96,0.22,0.2112>,
<0.98,0.22,0.2156>,
<1,0.22,0.22>,
<0,0.24,0>,
<0.02,0.24,0.0048>,
<0.04,0.24,0.0096>,
<0.06,0.24,0.0144>,
<0.08,0.24,0.0192>,
<0.1,0.24,0.024>,
<0.12,0.24,0.0288>,
<0.14,0.24,0.0336>,

<0.16,0.24,0.0384>,
<0.18,0.24,0.0432>,
<0.2,0.24,0.048>,
<0.22,0.24,0.0528>,
<0.24,0.24,0.0576>,
<0.26,0.24,0.0624>,
<0.28,0.24,0.0672>,
<0.3,0.24,0.072>,
<0.32,0.24,0.0768>,
<0.34,0.24,0.0816>,
<0.36,0.24,0.0864>,
<0.38,0.24,0.0912>,
<0.4,0.24,0.096>,
<0.42,0.24,0.1008>,
<0.44,0.24,0.1056>,
<0.46,0.24,0.1104>,
<0.48,0.24,0.1152>,
<0.5,0.24,0.12>,
<0.52,0.24,0.1248>,
<0.54,0.24,0.1296>,
<0.56,0.24,0.1344>,
<0.58,0.24,0.1392>,
<0.6,0.24,0.144>,
<0.62,0.24,0.1488>,
<0.64,0.24,0.1536>,
<0.66,0.24,0.1584>,
<0.68,0.24,0.1632>,
<0.7,0.24,0.168>,
<0.72,0.24,0.1728>,
<0.74,0.24,0.1776>,
<0.76,0.24,0.1824>,
<0.78,0.24,0.1872>,
<0.8,0.24,0.192>,
<0.82,0.24,0.1968>,
<0.84,0.24,0.2016>,
<0.86,0.24,0.2064>,
<0.88,0.24,0.2112>,
<0.9,0.24,0.216>,
<0.92,0.24,0.2208>,
<0.94,0.24,0.2256>,
<0.96,0.24,0.2304>,
<0.98,0.24,0.2352>,
<1,0.24,0.24>,
<0,0.26,0>,
<0.02,0.26,0.0052>,
<0.04,0.26,0.0104>,
<0.06,0.26,0.0156>,
<0.08,0.26,0.0208>,
<0.1,0.26,0.026>,
<0.12,0.26,0.0312>,
<0.14,0.26,0.0364>,
<0.16,0.26,0.0416>,
<0.18,0.26,0.0468>,
<0.2,0.26,0.052>,
<0.22,0.26,0.0572>,
<0.24,0.26,0.0624>,

<0.26,0.26,0.0676>,
<0.28,0.26,0.0728>,
<0.3,0.26,0.078>,
<0.32,0.26,0.0832>,
<0.34,0.26,0.0884>,
<0.36,0.26,0.0936>,
<0.38,0.26,0.0988>,
<0.4,0.26,0.104>,
<0.42,0.26,0.1092>,
<0.44,0.26,0.1144>,
<0.46,0.26,0.1196>,
<0.48,0.26,0.1248>,
<0.5,0.26,0.13>,
<0.52,0.26,0.1352>,
<0.54,0.26,0.1404>,
<0.56,0.26,0.1456>,
<0.58,0.26,0.1508>,
<0.6,0.26,0.156>,
<0.62,0.26,0.1612>,
<0.64,0.26,0.1664>,
<0.66,0.26,0.1716>,
<0.68,0.26,0.1768>,
<0.7,0.26,0.182>,
<0.72,0.26,0.1872>,
<0.74,0.26,0.1924>,
<0.76,0.26,0.1976>,
<0.78,0.26,0.2028>,
<0.8,0.26,0.208>,
<0.82,0.26,0.2132>,
<0.84,0.26,0.2184>,
<0.86,0.26,0.2236>,
<0.88,0.26,0.2288>,
<0.9,0.26,0.234>,
<0.92,0.26,0.2392>,
<0.94,0.26,0.2444>,
<0.96,0.26,0.2496>,
<0.98,0.26,0.2548>,
<1,0.26,0.26>,
<0,0.28,0>,
<0.02,0.28,0.0056>,
<0.04,0.28,0.0112>,
<0.06,0.28,0.0168>,
<0.08,0.28,0.0224>,
<0.1,0.28,0.028>,
<0.12,0.28,0.0336>,
<0.14,0.28,0.0392>,
<0.16,0.28,0.0448>,
<0.18,0.28,0.0504>,
<0.2,0.28,0.056>,
<0.22,0.28,0.0616>,
<0.24,0.28,0.0672>,
<0.26,0.28,0.0728>,
<0.28,0.28,0.0784>,
<0.3,0.28,0.084>,
<0.32,0.28,0.0896>,
<0.34,0.28,0.0952>,

<0.36,0.28,0.1008>,
<0.38,0.28,0.1064>,
<0.4,0.28,0.112>,
<0.42,0.28,0.1176>,
<0.44,0.28,0.1232>,
<0.46,0.28,0.1288>,
<0.48,0.28,0.1344>,
<0.5,0.28,0.14>,
<0.52,0.28,0.1456>,
<0.54,0.28,0.1512>,
<0.56,0.28,0.1568>,
<0.58,0.28,0.1624>,
<0.6,0.28,0.168>,
<0.62,0.28,0.1736>,
<0.64,0.28,0.1792>,
<0.66,0.28,0.1848>,
<0.68,0.28,0.1904>,
<0.7,0.28,0.196>,
<0.72,0.28,0.2016>,
<0.74,0.28,0.2072>,
<0.76,0.28,0.2128>,
<0.78,0.28,0.2184>,
<0.8,0.28,0.224>,
<0.82,0.28,0.2296>,
<0.84,0.28,0.2352>,
<0.86,0.28,0.2408>,
<0.88,0.28,0.2464>,
<0.9,0.28,0.252>,
<0.92,0.28,0.2576>,
<0.94,0.28,0.2632>,
<0.96,0.28,0.2688>,
<0.98,0.28,0.2744>,
<1,0.28,0.28>,
<0,0.3,0>,
<0.02,0.3,0.006>,
<0.04,0.3,0.012>,
<0.06,0.3,0.018>,
<0.08,0.3,0.024>,
<0.1,0.3,0.03>,
<0.12,0.3,0.036>,
<0.14,0.3,0.042>,
<0.16,0.3,0.048>,
<0.18,0.3,0.054>,
<0.2,0.3,0.06>,
<0.22,0.3,0.066>,
<0.24,0.3,0.072>,
<0.26,0.3,0.078>,
<0.28,0.3,0.084>,
<0.3,0.3,0.09>,
<0.32,0.3,0.096>,
<0.34,0.3,0.102>,
<0.36,0.3,0.108>,
<0.38,0.3,0.114>,
<0.4,0.3,0.12>,
<0.42,0.3,0.126>,
<0.44,0.3,0.132>,

<0.46,0.3,0.138>,
<0.48,0.3,0.144>,
<0.5,0.3,0.15>,
<0.52,0.3,0.156>,
<0.54,0.3,0.162>,
<0.56,0.3,0.168>,
<0.58,0.3,0.174>,
<0.6,0.3,0.18>,
<0.62,0.3,0.186>,
<0.64,0.3,0.192>,
<0.66,0.3,0.198>,
<0.68,0.3,0.204>,
<0.7,0.3,0.21>,
<0.72,0.3,0.216>,
<0.74,0.3,0.222>,
<0.76,0.3,0.228>,
<0.78,0.3,0.234>,
<0.8,0.3,0.24>,
<0.82,0.3,0.246>,
<0.84,0.3,0.252>,
<0.86,0.3,0.258>,
<0.88,0.3,0.264>,
<0.9,0.3,0.27>,
<0.92,0.3,0.276>,
<0.94,0.3,0.282>,
<0.96,0.3,0.288>,
<0.98,0.3,0.294>,
<1,0.3,0.3>,
<0,0.32,0>,
<0.02,0.32,0.0064>,
<0.04,0.32,0.0128>,
<0.06,0.32,0.0192>,
<0.08,0.32,0.0256>,
<0.1,0.32,0.032>,
<0.12,0.32,0.0384>,
<0.14,0.32,0.0448>,
<0.16,0.32,0.0512>,
<0.18,0.32,0.0576>,
<0.2,0.32,0.064>,
<0.22,0.32,0.0704>,
<0.24,0.32,0.0768>,
<0.26,0.32,0.0832>,
<0.28,0.32,0.0896>,
<0.3,0.32,0.096>,
<0.32,0.32,0.1024>,
<0.34,0.32,0.1088>,
<0.36,0.32,0.1152>,
<0.38,0.32,0.1216>,
<0.4,0.32,0.128>,
<0.42,0.32,0.1344>,
<0.44,0.32,0.1408>,
<0.46,0.32,0.1472>,
<0.48,0.32,0.1536>,
<0.5,0.32,0.16>,
<0.52,0.32,0.1664>,
<0.54,0.32,0.1728>,

<0.56,0.32,0.1792>,
<0.58,0.32,0.1856>,
<0.6,0.32,0.192>,
<0.62,0.32,0.1984>,
<0.64,0.32,0.2048>,
<0.66,0.32,0.2112>,
<0.68,0.32,0.2176>,
<0.7,0.32,0.224>,
<0.72,0.32,0.2304>,
<0.74,0.32,0.2368>,
<0.76,0.32,0.2432>,
<0.78,0.32,0.2496>,
<0.8,0.32,0.256>,
<0.82,0.32,0.2624>,
<0.84,0.32,0.2688>,
<0.86,0.32,0.2752>,
<0.88,0.32,0.2816>,
<0.9,0.32,0.288>,
<0.92,0.32,0.2944>,
<0.94,0.32,0.3008>,
<0.96,0.32,0.3072>,
<0.98,0.32,0.3136>,
<1,0.32,0.32>,
<0,0.34,0>,
<0.02,0.34,0.0068>,
<0.04,0.34,0.0136>,
<0.06,0.34,0.0204>,
<0.08,0.34,0.0272>,
<0.1,0.34,0.034>,
<0.12,0.34,0.0408>,
<0.14,0.34,0.0476>,
<0.16,0.34,0.0544>,
<0.18,0.34,0.0612>,
<0.2,0.34,0.068>,
<0.22,0.34,0.0748>,
<0.24,0.34,0.0816>,
<0.26,0.34,0.0884>,
<0.28,0.34,0.0952>,
<0.3,0.34,0.102>,
<0.32,0.34,0.1088>,
<0.34,0.34,0.1156>,
<0.36,0.34,0.1224>,
<0.38,0.34,0.1292>,
<0.4,0.34,0.136>,
<0.42,0.34,0.1428>,
<0.44,0.34,0.1496>,
<0.46,0.34,0.1564>,
<0.48,0.34,0.1632>,
<0.5,0.34,0.17>,
<0.52,0.34,0.1768>,
<0.54,0.34,0.1836>,
<0.56,0.34,0.1904>,
<0.58,0.34,0.1972>,
<0.6,0.34,0.204>,
<0.62,0.34,0.2108>,
<0.64,0.34,0.2176>,

<0.66,0.34,0.2244>,
<0.68,0.34,0.2312>,
<0.7,0.34,0.238>,
<0.72,0.34,0.2448>,
<0.74,0.34,0.2516>,
<0.76,0.34,0.2584>,
<0.78,0.34,0.2652>,
<0.8,0.34,0.272>,
<0.82,0.34,0.2788>,
<0.84,0.34,0.2856>,
<0.86,0.34,0.2924>,
<0.88,0.34,0.2992>,
<0.9,0.34,0.306>,
<0.92,0.34,0.3128>,
<0.94,0.34,0.3196>,
<0.96,0.34,0.3264>,
<0.98,0.34,0.3332>,
<1,0.34,0.34>,
<0,0.36,0>,
<0.02,0.36,0.0072>,
<0.04,0.36,0.0144>,
<0.06,0.36,0.0216>,
<0.08,0.36,0.0288>,
<0.1,0.36,0.036>,
<0.12,0.36,0.0432>,
<0.14,0.36,0.0504>,
<0.16,0.36,0.0576>,
<0.18,0.36,0.0648>,
<0.2,0.36,0.072>,
<0.22,0.36,0.0792>,
<0.24,0.36,0.0864>,
<0.26,0.36,0.0936>,
<0.28,0.36,0.1008>,
<0.3,0.36,0.108>,
<0.32,0.36,0.1152>,
<0.34,0.36,0.1224>,
<0.36,0.36,0.1296>,
<0.38,0.36,0.1368>,
<0.4,0.36,0.144>,
<0.42,0.36,0.1512>,
<0.44,0.36,0.1584>,
<0.46,0.36,0.1656>,
<0.48,0.36,0.1728>,
<0.5,0.36,0.18>,
<0.52,0.36,0.1872>,
<0.54,0.36,0.1944>,
<0.56,0.36,0.2016>,
<0.58,0.36,0.2088>,
<0.6,0.36,0.216>,
<0.62,0.36,0.2232>,
<0.64,0.36,0.2304>,
<0.66,0.36,0.2376>,
<0.68,0.36,0.2448>,
<0.7,0.36,0.252>,
<0.72,0.36,0.2592>,
<0.74,0.36,0.2664>,

<0.76,0.36,0.2736>,
<0.78,0.36,0.2808>,
<0.8,0.36,0.288>,
<0.82,0.36,0.2952>,
<0.84,0.36,0.3024>,
<0.86,0.36,0.3096>,
<0.88,0.36,0.3168>,
<0.9,0.36,0.324>,
<0.92,0.36,0.3312>,
<0.94,0.36,0.3384>,
<0.96,0.36,0.3456>,
<0.98,0.36,0.3528>,
<1,0.36,0.36>,
<0,0.38,0>,
<0.02,0.38,0.0076>,
<0.04,0.38,0.0152>,
<0.06,0.38,0.0228>,
<0.08,0.38,0.0304>,
<0.1,0.38,0.038>,
<0.12,0.38,0.0456>,
<0.14,0.38,0.0532>,
<0.16,0.38,0.0608>,
<0.18,0.38,0.0684>,
<0.2,0.38,0.076>,
<0.22,0.38,0.0836>,
<0.24,0.38,0.0912>,
<0.26,0.38,0.0988>,
<0.28,0.38,0.1064>,
<0.3,0.38,0.114>,
<0.32,0.38,0.1216>,
<0.34,0.38,0.1292>,
<0.36,0.38,0.1368>,
<0.38,0.38,0.1444>,
<0.4,0.38,0.152>,
<0.42,0.38,0.1596>,
<0.44,0.38,0.1672>,
<0.46,0.38,0.1748>,
<0.48,0.38,0.1824>,
<0.5,0.38,0.19>,
<0.52,0.38,0.1976>,
<0.54,0.38,0.2052>,
<0.56,0.38,0.2128>,
<0.58,0.38,0.2204>,
<0.6,0.38,0.228>,
<0.62,0.38,0.2356>,
<0.64,0.38,0.2432>,
<0.66,0.38,0.2508>,
<0.68,0.38,0.2584>,
<0.7,0.38,0.266>,
<0.72,0.38,0.2736>,
<0.74,0.38,0.2812>,
<0.76,0.38,0.2888>,
<0.78,0.38,0.2964>,
<0.8,0.38,0.304>,
<0.82,0.38,0.3116>,
<0.84,0.38,0.3192>,

<0.86,0.38,0.3268>,
<0.88,0.38,0.3344>,
<0.9,0.38,0.342>,
<0.92,0.38,0.3496>,
<0.94,0.38,0.3572>,
<0.96,0.38,0.3648>,
<0.98,0.38,0.3724>,
<1,0.38,0.38>,
<0,0.4,0>,
<0.02,0.4,0.008>,
<0.04,0.4,0.016>,
<0.06,0.4,0.024>,
<0.08,0.4,0.032>,
<0.1,0.4,0.04>,
<0.12,0.4,0.048>,
<0.14,0.4,0.056>,
<0.16,0.4,0.064>,
<0.18,0.4,0.072>,
<0.2,0.4,0.08>,
<0.22,0.4,0.088>,
<0.24,0.4,0.096>,
<0.26,0.4,0.104>,
<0.28,0.4,0.112>,
<0.3,0.4,0.12>,
<0.32,0.4,0.128>,
<0.34,0.4,0.136>,
<0.36,0.4,0.144>,
<0.38,0.4,0.152>,
<0.4,0.4,0.16>,
<0.42,0.4,0.168>,
<0.44,0.4,0.176>,
<0.46,0.4,0.184>,
<0.48,0.4,0.192>,
<0.5,0.4,0.2>,
<0.52,0.4,0.208>,
<0.54,0.4,0.216>,
<0.56,0.4,0.224>,
<0.58,0.4,0.232>,
<0.6,0.4,0.24>,
<0.62,0.4,0.248>,
<0.64,0.4,0.256>,
<0.66,0.4,0.264>,
<0.68,0.4,0.272>,
<0.7,0.4,0.28>,
<0.72,0.4,0.288>,
<0.74,0.4,0.296>,
<0.76,0.4,0.304>,
<0.78,0.4,0.312>,
<0.8,0.4,0.32>,
<0.82,0.4,0.328>,
<0.84,0.4,0.336>,
<0.86,0.4,0.344>,
<0.88,0.4,0.352>,
<0.9,0.4,0.36>,
<0.92,0.4,0.368>,
<0.94,0.4,0.376>,

<0.96,0.4,0.384>,
<0.98,0.4,0.392>,
<1,0.4,0.4>,
<0,0.42,0>,
<0.02,0.42,0.0084>,
<0.04,0.42,0.0168>,
<0.06,0.42,0.0252>,
<0.08,0.42,0.0336>,
<0.1,0.42,0.042>,
<0.12,0.42,0.0504>,
<0.14,0.42,0.0588>,
<0.16,0.42,0.0672>,
<0.18,0.42,0.0756>,
<0.2,0.42,0.084>,
<0.22,0.42,0.0924>,
<0.24,0.42,0.1008>,
<0.26,0.42,0.1092>,
<0.28,0.42,0.1176>,
<0.3,0.42,0.126>,
<0.32,0.42,0.1344>,
<0.34,0.42,0.1428>,
<0.36,0.42,0.1512>,
<0.38,0.42,0.1596>,
<0.4,0.42,0.168>,
<0.42,0.42,0.1764>,
<0.44,0.42,0.1848>,
<0.46,0.42,0.1932>,
<0.48,0.42,0.2016>,
<0.5,0.42,0.21>,
<0.52,0.42,0.2184>,
<0.54,0.42,0.2268>,
<0.56,0.42,0.2352>,
<0.58,0.42,0.2436>,
<0.6,0.42,0.252>,
<0.62,0.42,0.2604>,
<0.64,0.42,0.2688>,
<0.66,0.42,0.2772>,
<0.68,0.42,0.2856>,
<0.7,0.42,0.294>,
<0.72,0.42,0.3024>,
<0.74,0.42,0.3108>,
<0.76,0.42,0.3192>,
<0.78,0.42,0.3276>,
<0.8,0.42,0.336>,
<0.82,0.42,0.3444>,
<0.84,0.42,0.3528>,
<0.86,0.42,0.3612>,
<0.88,0.42,0.3696>,
<0.9,0.42,0.378>,
<0.92,0.42,0.3864>,
<0.94,0.42,0.3948>,
<0.96,0.42,0.4032>,
<0.98,0.42,0.4116>,
<1,0.42,0.42>,
<0,0.44,0>,
<0.02,0.44,0.0088>,

<0.04,0.44,0.0176>,
<0.06,0.44,0.0264>,
<0.08,0.44,0.0352>,
<0.1,0.44,0.044>,
<0.12,0.44,0.0528>,
<0.14,0.44,0.0616>,
<0.16,0.44,0.0704>,
<0.18,0.44,0.0792>,
<0.2,0.44,0.088>,
<0.22,0.44,0.0968>,
<0.24,0.44,0.1056>,
<0.26,0.44,0.1144>,
<0.28,0.44,0.1232>,
<0.3,0.44,0.132>,
<0.32,0.44,0.1408>,
<0.34,0.44,0.1496>,
<0.36,0.44,0.1584>,
<0.38,0.44,0.1672>,
<0.4,0.44,0.176>,
<0.42,0.44,0.1848>,
<0.44,0.44,0.1936>,
<0.46,0.44,0.2024>,
<0.48,0.44,0.2112>,
<0.5,0.44,0.22>,
<0.52,0.44,0.2288>,
<0.54,0.44,0.2376>,
<0.56,0.44,0.2464>,
<0.58,0.44,0.2552>,
<0.6,0.44,0.264>,
<0.62,0.44,0.2728>,
<0.64,0.44,0.2816>,
<0.66,0.44,0.2904>,
<0.68,0.44,0.2992>,
<0.7,0.44,0.308>,
<0.72,0.44,0.3168>,
<0.74,0.44,0.3256>,
<0.76,0.44,0.3344>,
<0.78,0.44,0.3432>,
<0.8,0.44,0.352>,
<0.82,0.44,0.3608>,
<0.84,0.44,0.3696>,
<0.86,0.44,0.3784>,
<0.88,0.44,0.3872>,
<0.9,0.44,0.396>,
<0.92,0.44,0.4048>,
<0.94,0.44,0.4136>,
<0.96,0.44,0.4224>,
<0.98,0.44,0.4312>,
<1,0.44,0.44>,
<0,0.46,0>,
<0.02,0.46,0.0092>,
<0.04,0.46,0.0184>,
<0.06,0.46,0.0276>,
<0.08,0.46,0.0368>,
<0.1,0.46,0.046>,
<0.12,0.46,0.0552>,

<0.14,0.46,0.0644>,
<0.16,0.46,0.0736>,
<0.18,0.46,0.0828>,
<0.2,0.46,0.092>,
<0.22,0.46,0.1012>,
<0.24,0.46,0.1104>,
<0.26,0.46,0.1196>,
<0.28,0.46,0.1288>,
<0.3,0.46,0.138>,
<0.32,0.46,0.1472>,
<0.34,0.46,0.1564>,
<0.36,0.46,0.1656>,
<0.38,0.46,0.1748>,
<0.4,0.46,0.184>,
<0.42,0.46,0.1932>,
<0.44,0.46,0.2024>,
<0.46,0.46,0.2116>,
<0.48,0.46,0.2208>,
<0.5,0.46,0.23>,
<0.52,0.46,0.2392>,
<0.54,0.46,0.2484>,
<0.56,0.46,0.2576>,
<0.58,0.46,0.2668>,
<0.6,0.46,0.276>,
<0.62,0.46,0.2852>,
<0.64,0.46,0.2944>,
<0.66,0.46,0.3036>,
<0.68,0.46,0.3128>,
<0.7,0.46,0.322>,
<0.72,0.46,0.3312>,
<0.74,0.46,0.3404>,
<0.76,0.46,0.3496>,
<0.78,0.46,0.3588>,
<0.8,0.46,0.368>,
<0.82,0.46,0.3772>,
<0.84,0.46,0.3864>,
<0.86,0.46,0.3956>,
<0.88,0.46,0.4048>,
<0.9,0.46,0.414>,
<0.92,0.46,0.4232>,
<0.94,0.46,0.4324>,
<0.96,0.46,0.4416>,
<0.98,0.46,0.4508>,
<1,0.46,0.46>,
<0,0.48,0>,
<0.02,0.48,0.0096>,
<0.04,0.48,0.0192>,
<0.06,0.48,0.0288>,
<0.08,0.48,0.0384>,
<0.1,0.48,0.048>,
<0.12,0.48,0.0576>,
<0.14,0.48,0.0672>,
<0.16,0.48,0.0768>,
<0.18,0.48,0.0864>,
<0.2,0.48,0.096>,
<0.22,0.48,0.1056>,

<0.24,0.48,0.1152>,
<0.26,0.48,0.1248>,
<0.28,0.48,0.1344>,
<0.3,0.48,0.144>,
<0.32,0.48,0.1536>,
<0.34,0.48,0.1632>,
<0.36,0.48,0.1728>,
<0.38,0.48,0.1824>,
<0.4,0.48,0.192>,
<0.42,0.48,0.2016>,
<0.44,0.48,0.2112>,
<0.46,0.48,0.2208>,
<0.48,0.48,0.2304>,
<0.5,0.48,0.24>,
<0.52,0.48,0.2496>,
<0.54,0.48,0.2592>,
<0.56,0.48,0.2688>,
<0.58,0.48,0.2784>,
<0.6,0.48,0.288>,
<0.62,0.48,0.2976>,
<0.64,0.48,0.3072>,
<0.66,0.48,0.3168>,
<0.68,0.48,0.3264>,
<0.7,0.48,0.336>,
<0.72,0.48,0.3456>,
<0.74,0.48,0.3552>,
<0.76,0.48,0.3648>,
<0.78,0.48,0.3744>,
<0.8,0.48,0.384>,
<0.82,0.48,0.3936>,
<0.84,0.48,0.4032>,
<0.86,0.48,0.4128>,
<0.88,0.48,0.4224>,
<0.9,0.48,0.432>,
<0.92,0.48,0.4416>,
<0.94,0.48,0.4512>,
<0.96,0.48,0.4608>,
<0.98,0.48,0.4704>,
<1,0.48,0.48>,
<0,0.5,0>,
<0.02,0.5,0.01>,
<0.04,0.5,0.02>,
<0.06,0.5,0.03>,
<0.08,0.5,0.04>,
<0.1,0.5,0.05>,
<0.12,0.5,0.06>,
<0.14,0.5,0.07>,
<0.16,0.5,0.08>,
<0.18,0.5,0.09>,
<0.2,0.5,0.1>,
<0.22,0.5,0.11>,
<0.24,0.5,0.12>,
<0.26,0.5,0.13>,
<0.28,0.5,0.14>,
<0.3,0.5,0.15>,
<0.32,0.5,0.16>,

<0.34,0.5,0.17>,
<0.36,0.5,0.18>,
<0.38,0.5,0.19>,
<0.4,0.5,0.2>,
<0.42,0.5,0.21>,
<0.44,0.5,0.22>,
<0.46,0.5,0.23>,
<0.48,0.5,0.24>,
<0.5,0.5,0.25>,
<0.52,0.5,0.26>,
<0.54,0.5,0.27>,
<0.56,0.5,0.28>,
<0.58,0.5,0.29>,
<0.6,0.5,0.3>,
<0.62,0.5,0.31>,
<0.64,0.5,0.32>,
<0.66,0.5,0.33>,
<0.68,0.5,0.34>,
<0.7,0.5,0.35>,
<0.72,0.5,0.36>,
<0.74,0.5,0.37>,
<0.76,0.5,0.38>,
<0.78,0.5,0.39>,
<0.8,0.5,0.4>,
<0.82,0.5,0.41>,
<0.84,0.5,0.42>,
<0.86,0.5,0.43>,
<0.88,0.5,0.44>,
<0.9,0.5,0.45>,
<0.92,0.5,0.46>,
<0.94,0.5,0.47>,
<0.96,0.5,0.48>,
<0.98,0.5,0.49>,
<1,0.5,0.5>,
<0,0.52,0>,
<0.02,0.52,0.0104>,
<0.04,0.52,0.0208>,
<0.06,0.52,0.0312>,
<0.08,0.52,0.0416>,
<0.1,0.52,0.052>,
<0.12,0.52,0.0624>,
<0.14,0.52,0.0728>,
<0.16,0.52,0.0832>,
<0.18,0.52,0.0936>,
<0.2,0.52,0.104>,
<0.22,0.52,0.1144>,
<0.24,0.52,0.1248>,
<0.26,0.52,0.1352>,
<0.28,0.52,0.1456>,
<0.3,0.52,0.156>,
<0.32,0.52,0.1664>,
<0.34,0.52,0.1768>,
<0.36,0.52,0.1872>,
<0.38,0.52,0.1976>,
<0.4,0.52,0.208>,
<0.42,0.52,0.2184>,

<0.44,0.52,0.2288>,
<0.46,0.52,0.2392>,
<0.48,0.52,0.2496>,
<0.5,0.52,0.26>,
<0.52,0.52,0.2704>,
<0.54,0.52,0.2808>,
<0.56,0.52,0.2912>,
<0.58,0.52,0.3016>,
<0.6,0.52,0.312>,
<0.62,0.52,0.3224>,
<0.64,0.52,0.3328>,
<0.66,0.52,0.3432>,
<0.68,0.52,0.3536>,
<0.7,0.52,0.364>,
<0.72,0.52,0.3744>,
<0.74,0.52,0.3848>,
<0.76,0.52,0.3952>,
<0.78,0.52,0.4056>,
<0.8,0.52,0.416>,
<0.82,0.52,0.4264>,
<0.84,0.52,0.4368>,
<0.86,0.52,0.4472>,
<0.88,0.52,0.4576>,
<0.9,0.52,0.468>,
<0.92,0.52,0.4784>,
<0.94,0.52,0.4888>,
<0.96,0.52,0.4992>,
<0.98,0.52,0.5096>,
<1,0.52,0.52>,
<0,0.54,0>,
<0.02,0.54,0.0108>,
<0.04,0.54,0.0216>,
<0.06,0.54,0.0324>,
<0.08,0.54,0.0432>,
<0.1,0.54,0.054>,
<0.12,0.54,0.0648>,
<0.14,0.54,0.0756>,
<0.16,0.54,0.0864>,
<0.18,0.54,0.0972>,
<0.2,0.54,0.108>,
<0.22,0.54,0.1188>,
<0.24,0.54,0.1296>,
<0.26,0.54,0.1404>,
<0.28,0.54,0.1512>,
<0.3,0.54,0.162>,
<0.32,0.54,0.1728>,
<0.34,0.54,0.1836>,
<0.36,0.54,0.1944>,
<0.38,0.54,0.2052>,
<0.4,0.54,0.216>,
<0.42,0.54,0.2268>,
<0.44,0.54,0.2376>,
<0.46,0.54,0.2484>,
<0.48,0.54,0.2592>,
<0.5,0.54,0.27>,
<0.52,0.54,0.2808>,

<0.54,0.54,0.2916>,
<0.56,0.54,0.3024>,
<0.58,0.54,0.3132>,
<0.6,0.54,0.324>,
<0.62,0.54,0.3348>,
<0.64,0.54,0.3456>,
<0.66,0.54,0.3564>,
<0.68,0.54,0.3672>,
<0.7,0.54,0.378>,
<0.72,0.54,0.3888>,
<0.74,0.54,0.3996>,
<0.76,0.54,0.4104>,
<0.78,0.54,0.4212>,
<0.8,0.54,0.432>,
<0.82,0.54,0.4428>,
<0.84,0.54,0.4536>,
<0.86,0.54,0.4644>,
<0.88,0.54,0.4752>,
<0.9,0.54,0.486>,
<0.92,0.54,0.4968>,
<0.94,0.54,0.5076>,
<0.96,0.54,0.5184>,
<0.98,0.54,0.5292>,
<1,0.54,0.54>,
<0,0.56,0>,
<0.02,0.56,0.0112>,
<0.04,0.56,0.0224>,
<0.06,0.56,0.0336>,
<0.08,0.56,0.0448>,
<0.1,0.56,0.056>,
<0.12,0.56,0.0672>,
<0.14,0.56,0.0784>,
<0.16,0.56,0.0896>,
<0.18,0.56,0.1008>,
<0.2,0.56,0.112>,
<0.22,0.56,0.1232>,
<0.24,0.56,0.1344>,
<0.26,0.56,0.1456>,
<0.28,0.56,0.1568>,
<0.3,0.56,0.168>,
<0.32,0.56,0.1792>,
<0.34,0.56,0.1904>,
<0.36,0.56,0.2016>,
<0.38,0.56,0.2128>,
<0.4,0.56,0.224>,
<0.42,0.56,0.2352>,
<0.44,0.56,0.2464>,
<0.46,0.56,0.2576>,
<0.48,0.56,0.2688>,
<0.5,0.56,0.28>,
<0.52,0.56,0.2912>,
<0.54,0.56,0.3024>,
<0.56,0.56,0.3136>,
<0.58,0.56,0.3248>,
<0.6,0.56,0.336>,
<0.62,0.56,0.3472>,

<0.64,0.56,0.3584>,
<0.66,0.56,0.3696>,
<0.68,0.56,0.3808>,
<0.7,0.56,0.392>,
<0.72,0.56,0.4032>,
<0.74,0.56,0.4144>,
<0.76,0.56,0.4256>,
<0.78,0.56,0.4368>,
<0.8,0.56,0.448>,
<0.82,0.56,0.4592>,
<0.84,0.56,0.4704>,
<0.86,0.56,0.4816>,
<0.88,0.56,0.4928>,
<0.9,0.56,0.504>,
<0.92,0.56,0.5152>,
<0.94,0.56,0.5264>,
<0.96,0.56,0.5376>,
<0.98,0.56,0.5488>,
<1,0.56,0.56>,
<0,0.58,0>,
<0.02,0.58,0.0116>,
<0.04,0.58,0.0232>,
<0.06,0.58,0.0348>,
<0.08,0.58,0.0464>,
<0.1,0.58,0.058>,
<0.12,0.58,0.0696>,
<0.14,0.58,0.0812>,
<0.16,0.58,0.0928>,
<0.18,0.58,0.1044>,
<0.2,0.58,0.116>,
<0.22,0.58,0.1276>,
<0.24,0.58,0.1392>,
<0.26,0.58,0.1508>,
<0.28,0.58,0.1624>,
<0.3,0.58,0.174>,
<0.32,0.58,0.1856>,
<0.34,0.58,0.1972>,
<0.36,0.58,0.2088>,
<0.38,0.58,0.2204>,
<0.4,0.58,0.232>,
<0.42,0.58,0.2436>,
<0.44,0.58,0.2552>,
<0.46,0.58,0.2668>,
<0.48,0.58,0.2784>,
<0.5,0.58,0.29>,
<0.52,0.58,0.3016>,
<0.54,0.58,0.3132>,
<0.56,0.58,0.3248>,
<0.58,0.58,0.3364>,
<0.6,0.58,0.348>,
<0.62,0.58,0.3596>,
<0.64,0.58,0.3712>,
<0.66,0.58,0.3828>,
<0.68,0.58,0.3944>,
<0.7,0.58,0.406>,
<0.72,0.58,0.4176>,

<0.74,0.58,0.4292>,
<0.76,0.58,0.4408>,
<0.78,0.58,0.4524>,
<0.8,0.58,0.464>,
<0.82,0.58,0.4756>,
<0.84,0.58,0.4872>,
<0.86,0.58,0.4988>,
<0.88,0.58,0.5104>,
<0.9,0.58,0.522>,
<0.92,0.58,0.5336>,
<0.94,0.58,0.5452>,
<0.96,0.58,0.5568>,
<0.98,0.58,0.5684>,
<1,0.58,0.58>,
<0,0.6,0>,
<0.02,0.6,0.012>,
<0.04,0.6,0.024>,
<0.06,0.6,0.036>,
<0.08,0.6,0.048>,
<0.1,0.6,0.06>,
<0.12,0.6,0.072>,
<0.14,0.6,0.084>,
<0.16,0.6,0.096>,
<0.18,0.6,0.108>,
<0.2,0.6,0.12>,
<0.22,0.6,0.132>,
<0.24,0.6,0.144>,
<0.26,0.6,0.156>,
<0.28,0.6,0.168>,
<0.3,0.6,0.18>,
<0.32,0.6,0.192>,
<0.34,0.6,0.204>,
<0.36,0.6,0.216>,
<0.38,0.6,0.228>,
<0.4,0.6,0.24>,
<0.42,0.6,0.252>,
<0.44,0.6,0.264>,
<0.46,0.6,0.276>,
<0.48,0.6,0.288>,
<0.5,0.6,0.3>,
<0.52,0.6,0.312>,
<0.54,0.6,0.324>,
<0.56,0.6,0.336>,
<0.58,0.6,0.348>,
<0.6,0.6,0.36>,
<0.62,0.6,0.372>,
<0.64,0.6,0.384>,
<0.66,0.6,0.396>,
<0.68,0.6,0.408>,
<0.7,0.6,0.42>,
<0.72,0.6,0.432>,
<0.74,0.6,0.444>,
<0.76,0.6,0.456>,
<0.78,0.6,0.468>,
<0.8,0.6,0.48>,
<0.82,0.6,0.492>,

<0.84,0.6,0.504>,
<0.86,0.6,0.516>,
<0.88,0.6,0.528>,
<0.9,0.6,0.54>,
<0.92,0.6,0.552>,
<0.94,0.6,0.564>,
<0.96,0.6,0.576>,
<0.98,0.6,0.588>,
<1,0.6,0.6>,
<0,0.62,0>,
<0.02,0.62,0.0124>,
<0.04,0.62,0.0248>,
<0.06,0.62,0.0372>,
<0.08,0.62,0.0496>,
<0.1,0.62,0.062>,
<0.12,0.62,0.0744>,
<0.14,0.62,0.0868>,
<0.16,0.62,0.0992>,
<0.18,0.62,0.1116>,
<0.2,0.62,0.124>,
<0.22,0.62,0.1364>,
<0.24,0.62,0.1488>,
<0.26,0.62,0.1612>,
<0.28,0.62,0.1736>,
<0.3,0.62,0.186>,
<0.32,0.62,0.1984>,
<0.34,0.62,0.2108>,
<0.36,0.62,0.2232>,
<0.38,0.62,0.2356>,
<0.4,0.62,0.248>,
<0.42,0.62,0.2604>,
<0.44,0.62,0.2728>,
<0.46,0.62,0.2852>,
<0.48,0.62,0.2976>,
<0.5,0.62,0.31>,
<0.52,0.62,0.3224>,
<0.54,0.62,0.3348>,
<0.56,0.62,0.3472>,
<0.58,0.62,0.3596>,
<0.6,0.62,0.372>,
<0.62,0.62,0.3844>,
<0.64,0.62,0.3968>,
<0.66,0.62,0.4092>,
<0.68,0.62,0.4216>,
<0.7,0.62,0.434>,
<0.72,0.62,0.4464>,
<0.74,0.62,0.4588>,
<0.76,0.62,0.4712>,
<0.78,0.62,0.4836>,
<0.8,0.62,0.496>,
<0.82,0.62,0.5084>,
<0.84,0.62,0.5208>,
<0.86,0.62,0.5332>,
<0.88,0.62,0.5456>,
<0.9,0.62,0.558>,
<0.92,0.62,0.5704>,

<0.94,0.62,0.5828>,
<0.96,0.62,0.5952>,
<0.98,0.62,0.6076>,
<1,0.62,0.62>,
<0,0.64,0>,
<0.02,0.64,0.0128>,
<0.04,0.64,0.0256>,
<0.06,0.64,0.0384>,
<0.08,0.64,0.0512>,
<0.1,0.64,0.064>,
<0.12,0.64,0.0768>,
<0.14,0.64,0.0896>,
<0.16,0.64,0.1024>,
<0.18,0.64,0.1152>,
<0.2,0.64,0.128>,
<0.22,0.64,0.1408>,
<0.24,0.64,0.1536>,
<0.26,0.64,0.1664>,
<0.28,0.64,0.1792>,
<0.3,0.64,0.192>,
<0.32,0.64,0.2048>,
<0.34,0.64,0.2176>,
<0.36,0.64,0.2304>,
<0.38,0.64,0.2432>,
<0.4,0.64,0.256>,
<0.42,0.64,0.2688>,
<0.44,0.64,0.2816>,
<0.46,0.64,0.2944>,
<0.48,0.64,0.3072>,
<0.5,0.64,0.32>,
<0.52,0.64,0.3328>,
<0.54,0.64,0.3456>,
<0.56,0.64,0.3584>,
<0.58,0.64,0.3712>,
<0.6,0.64,0.384>,
<0.62,0.64,0.3968>,
<0.64,0.64,0.4096>,
<0.66,0.64,0.4224>,
<0.68,0.64,0.4352>,
<0.7,0.64,0.448>,
<0.72,0.64,0.4608>,
<0.74,0.64,0.4736>,
<0.76,0.64,0.4864>,
<0.78,0.64,0.4992>,
<0.8,0.64,0.512>,
<0.82,0.64,0.5248>,
<0.84,0.64,0.5376>,
<0.86,0.64,0.5504>,
<0.88,0.64,0.5632>,
<0.9,0.64,0.576>,
<0.92,0.64,0.5888>,
<0.94,0.64,0.6016>,
<0.96,0.64,0.6144>,
<0.98,0.64,0.6272>,
<1,0.64,0.64>,
<0,0.66,0>,

<0.02,0.66,0.0132>,
<0.04,0.66,0.0264>,
<0.06,0.66,0.0396>,
<0.08,0.66,0.0528>,
<0.1,0.66,0.066>,
<0.12,0.66,0.0792>,
<0.14,0.66,0.0924>,
<0.16,0.66,0.1056>,
<0.18,0.66,0.1188>,
<0.2,0.66,0.132>,
<0.22,0.66,0.1452>,
<0.24,0.66,0.1584>,
<0.26,0.66,0.1716>,
<0.28,0.66,0.1848>,
<0.3,0.66,0.198>,
<0.32,0.66,0.2112>,
<0.34,0.66,0.2244>,
<0.36,0.66,0.2376>,
<0.38,0.66,0.2508>,
<0.4,0.66,0.264>,
<0.42,0.66,0.2772>,
<0.44,0.66,0.2904>,
<0.46,0.66,0.3036>,
<0.48,0.66,0.3168>,
<0.5,0.66,0.33>,
<0.52,0.66,0.3432>,
<0.54,0.66,0.3564>,
<0.56,0.66,0.3696>,
<0.58,0.66,0.3828>,
<0.6,0.66,0.396>,
<0.62,0.66,0.4092>,
<0.64,0.66,0.4224>,
<0.66,0.66,0.4356>,
<0.68,0.66,0.4488>,
<0.7,0.66,0.462>,
<0.72,0.66,0.4752>,
<0.74,0.66,0.4884>,
<0.76,0.66,0.5016>,
<0.78,0.66,0.5148>,
<0.8,0.66,0.528>,
<0.82,0.66,0.5412>,
<0.84,0.66,0.5544>,
<0.86,0.66,0.5676>,
<0.88,0.66,0.5808>,
<0.9,0.66,0.594>,
<0.92,0.66,0.6072>,
<0.94,0.66,0.6204>,
<0.96,0.66,0.6336>,
<0.98,0.66,0.6468>,
<1,0.66,0.66>,
<0,0.68,0>,
<0.02,0.68,0.0136>,
<0.04,0.68,0.0272>,
<0.06,0.68,0.0408>,
<0.08,0.68,0.0544>,
<0.1,0.68,0.068>,

<0.12,0.68,0.0816>,
<0.14,0.68,0.0952>,
<0.16,0.68,0.1088>,
<0.18,0.68,0.1224>,
<0.2,0.68,0.136>,
<0.22,0.68,0.1496>,
<0.24,0.68,0.1632>,
<0.26,0.68,0.1768>,
<0.28,0.68,0.1904>,
<0.3,0.68,0.204>,
<0.32,0.68,0.2176>,
<0.34,0.68,0.2312>,
<0.36,0.68,0.2448>,
<0.38,0.68,0.2584>,
<0.4,0.68,0.272>,
<0.42,0.68,0.2856>,
<0.44,0.68,0.2992>,
<0.46,0.68,0.3128>,
<0.48,0.68,0.3264>,
<0.5,0.68,0.34>,
<0.52,0.68,0.3536>,
<0.54,0.68,0.3672>,
<0.56,0.68,0.3808>,
<0.58,0.68,0.3944>,
<0.6,0.68,0.408>,
<0.62,0.68,0.4216>,
<0.64,0.68,0.4352>,
<0.66,0.68,0.4488>,
<0.68,0.68,0.4624>,
<0.7,0.68,0.476>,
<0.72,0.68,0.4896>,
<0.74,0.68,0.5032>,
<0.76,0.68,0.5168>,
<0.78,0.68,0.5304>,
<0.8,0.68,0.544>,
<0.82,0.68,0.5576>,
<0.84,0.68,0.5712>,
<0.86,0.68,0.5848>,
<0.88,0.68,0.5984>,
<0.9,0.68,0.612>,
<0.92,0.68,0.6256>,
<0.94,0.68,0.6392>,
<0.96,0.68,0.6528>,
<0.98,0.68,0.6664>,
<1,0.68,0.68>,
<0,0.7,0>,
<0.02,0.7,0.014>,
<0.04,0.7,0.028>,
<0.06,0.7,0.042>,
<0.08,0.7,0.056>,
<0.1,0.7,0.07>,
<0.12,0.7,0.084>,
<0.14,0.7,0.098>,
<0.16,0.7,0.112>,
<0.18,0.7,0.126>,
<0.2,0.7,0.14>,

<0.22,0.7,0.154>,
<0.24,0.7,0.168>,
<0.26,0.7,0.182>,
<0.28,0.7,0.196>,
<0.3,0.7,0.21>,
<0.32,0.7,0.224>,
<0.34,0.7,0.238>,
<0.36,0.7,0.252>,
<0.38,0.7,0.266>,
<0.4,0.7,0.28>,
<0.42,0.7,0.294>,
<0.44,0.7,0.308>,
<0.46,0.7,0.322>,
<0.48,0.7,0.336>,
<0.5,0.7,0.35>,
<0.52,0.7,0.364>,
<0.54,0.7,0.378>,
<0.56,0.7,0.392>,
<0.58,0.7,0.406>,
<0.6,0.7,0.42>,
<0.62,0.7,0.434>,
<0.64,0.7,0.448>,
<0.66,0.7,0.462>,
<0.68,0.7,0.476>,
<0.7,0.7,0.49>,
<0.72,0.7,0.504>,
<0.74,0.7,0.518>,
<0.76,0.7,0.532>,
<0.78,0.7,0.546>,
<0.8,0.7,0.56>,
<0.82,0.7,0.574>,
<0.84,0.7,0.588>,
<0.86,0.7,0.602>,
<0.88,0.7,0.616>,
<0.9,0.7,0.63>,
<0.92,0.7,0.644>,
<0.94,0.7,0.658>,
<0.96,0.7,0.672>,
<0.98,0.7,0.686>,
<1,0.7,0.7>,
<0,0.72,0>,
<0.02,0.72,0.0144>,
<0.04,0.72,0.0288>,
<0.06,0.72,0.0432>,
<0.08,0.72,0.0576>,
<0.1,0.72,0.072>,
<0.12,0.72,0.0864>,
<0.14,0.72,0.1008>,
<0.16,0.72,0.1152>,
<0.18,0.72,0.1296>,
<0.2,0.72,0.144>,
<0.22,0.72,0.1584>,
<0.24,0.72,0.1728>,
<0.26,0.72,0.1872>,
<0.28,0.72,0.2016>,
<0.3,0.72,0.216>,

<0.32,0.72,0.2304>,
<0.34,0.72,0.2448>,
<0.36,0.72,0.2592>,
<0.38,0.72,0.2736>,
<0.4,0.72,0.288>,
<0.42,0.72,0.3024>,
<0.44,0.72,0.3168>,
<0.46,0.72,0.3312>,
<0.48,0.72,0.3456>,
<0.5,0.72,0.36>,
<0.52,0.72,0.3744>,
<0.54,0.72,0.3888>,
<0.56,0.72,0.4032>,
<0.58,0.72,0.4176>,
<0.6,0.72,0.432>,
<0.62,0.72,0.4464>,
<0.64,0.72,0.4608>,
<0.66,0.72,0.4752>,
<0.68,0.72,0.4896>,
<0.7,0.72,0.504>,
<0.72,0.72,0.5184>,
<0.74,0.72,0.5328>,
<0.76,0.72,0.5472>,
<0.78,0.72,0.5616>,
<0.8,0.72,0.576>,
<0.82,0.72,0.5904>,
<0.84,0.72,0.6048>,
<0.86,0.72,0.6192>,
<0.88,0.72,0.6336>,
<0.9,0.72,0.648>,
<0.92,0.72,0.6624>,
<0.94,0.72,0.6768>,
<0.96,0.72,0.6912>,
<0.98,0.72,0.7056>,
<1,0.72,0.72>,
<0,0.74,0>,
<0.02,0.74,0.0148>,
<0.04,0.74,0.0296>,
<0.06,0.74,0.0444>,
<0.08,0.74,0.0592>,
<0.1,0.74,0.074>,
<0.12,0.74,0.0888>,
<0.14,0.74,0.1036>,
<0.16,0.74,0.1184>,
<0.18,0.74,0.1332>,
<0.2,0.74,0.148>,
<0.22,0.74,0.1628>,
<0.24,0.74,0.1776>,
<0.26,0.74,0.1924>,
<0.28,0.74,0.2072>,
<0.3,0.74,0.222>,
<0.32,0.74,0.2368>,
<0.34,0.74,0.2516>,
<0.36,0.74,0.2664>,
<0.38,0.74,0.2812>,
<0.4,0.74,0.296>,

<0.42,0.74,0.3108>,
<0.44,0.74,0.3256>,
<0.46,0.74,0.3404>,
<0.48,0.74,0.3552>,
<0.5,0.74,0.37>,
<0.52,0.74,0.3848>,
<0.54,0.74,0.3996>,
<0.56,0.74,0.4144>,
<0.58,0.74,0.4292>,
<0.6,0.74,0.4444>,
<0.62,0.74,0.4588>,
<0.64,0.74,0.4736>,
<0.66,0.74,0.4884>,
<0.68,0.74,0.5032>,
<0.7,0.74,0.518>,
<0.72,0.74,0.5328>,
<0.74,0.74,0.5476>,
<0.76,0.74,0.5624>,
<0.78,0.74,0.5772>,
<0.8,0.74,0.592>,
<0.82,0.74,0.6068>,
<0.84,0.74,0.6216>,
<0.86,0.74,0.6364>,
<0.88,0.74,0.6512>,
<0.9,0.74,0.666>,
<0.92,0.74,0.6808>,
<0.94,0.74,0.6956>,
<0.96,0.74,0.7104>,
<0.98,0.74,0.7252>,
<1,0.74,0.74>,
<0,0.76,0>,
<0.02,0.76,0.0152>,
<0.04,0.76,0.0304>,
<0.06,0.76,0.0456>,
<0.08,0.76,0.0608>,
<0.1,0.76,0.076>,
<0.12,0.76,0.0912>,
<0.14,0.76,0.1064>,
<0.16,0.76,0.1216>,
<0.18,0.76,0.1368>,
<0.2,0.76,0.152>,
<0.22,0.76,0.1672>,
<0.24,0.76,0.1824>,
<0.26,0.76,0.1976>,
<0.28,0.76,0.2128>,
<0.3,0.76,0.228>,
<0.32,0.76,0.2432>,
<0.34,0.76,0.2584>,
<0.36,0.76,0.2736>,
<0.38,0.76,0.2888>,
<0.4,0.76,0.304>,
<0.42,0.76,0.3192>,
<0.44,0.76,0.3344>,
<0.46,0.76,0.3496>,
<0.48,0.76,0.3648>,
<0.5,0.76,0.38>,

<0.52,0.76,0.3952>,
<0.54,0.76,0.4104>,
<0.56,0.76,0.4256>,
<0.58,0.76,0.4408>,
<0.6,0.76,0.456>,
<0.62,0.76,0.4712>,
<0.64,0.76,0.4864>,
<0.66,0.76,0.5016>,
<0.68,0.76,0.5168>,
<0.7,0.76,0.532>,
<0.72,0.76,0.5472>,
<0.74,0.76,0.5624>,
<0.76,0.76,0.5776>,
<0.78,0.76,0.5928>,
<0.8,0.76,0.608>,
<0.82,0.76,0.6232>,
<0.84,0.76,0.6384>,
<0.86,0.76,0.6536>,
<0.88,0.76,0.6688>,
<0.9,0.76,0.684>,
<0.92,0.76,0.6992>,
<0.94,0.76,0.7144>,
<0.96,0.76,0.7296>,
<0.98,0.76,0.7448>,
<1,0.76,0.76>,
<0,0.78,0>,
<0.02,0.78,0.0156>,
<0.04,0.78,0.0312>,
<0.06,0.78,0.0468>,
<0.08,0.78,0.0624>,
<0.1,0.78,0.078>,
<0.12,0.78,0.0936>,
<0.14,0.78,0.1092>,
<0.16,0.78,0.1248>,
<0.18,0.78,0.1404>,
<0.2,0.78,0.156>,
<0.22,0.78,0.1716>,
<0.24,0.78,0.1872>,
<0.26,0.78,0.2028>,
<0.28,0.78,0.2184>,
<0.3,0.78,0.234>,
<0.32,0.78,0.2496>,
<0.34,0.78,0.2652>,
<0.36,0.78,0.2808>,
<0.38,0.78,0.2964>,
<0.4,0.78,0.312>,
<0.42,0.78,0.3276>,
<0.44,0.78,0.3432>,
<0.46,0.78,0.3588>,
<0.48,0.78,0.3744>,
<0.5,0.78,0.39>,
<0.52,0.78,0.4056>,
<0.54,0.78,0.4212>,
<0.56,0.78,0.4368>,
<0.58,0.78,0.4524>,
<0.6,0.78,0.468>,

<0.62,0.78,0.4836>,
<0.64,0.78,0.4992>,
<0.66,0.78,0.5148>,
<0.68,0.78,0.5304>,
<0.7,0.78,0.546>,
<0.72,0.78,0.5616>,
<0.74,0.78,0.5772>,
<0.76,0.78,0.5928>,
<0.78,0.78,0.6084>,
<0.8,0.78,0.624>,
<0.82,0.78,0.6396>,
<0.84,0.78,0.6552>,
<0.86,0.78,0.6708>,
<0.88,0.78,0.6864>,
<0.9,0.78,0.702>,
<0.92,0.78,0.7176>,
<0.94,0.78,0.7332>,
<0.96,0.78,0.7488>,
<0.98,0.78,0.7644>,
<1,0.78,0.78>,
<0,0.8,0>,
<0.02,0.8,0.016>,
<0.04,0.8,0.032>,
<0.06,0.8,0.048>,
<0.08,0.8,0.064>,
<0.1,0.8,0.08>,
<0.12,0.8,0.096>,
<0.14,0.8,0.112>,
<0.16,0.8,0.128>,
<0.18,0.8,0.144>,
<0.2,0.8,0.16>,
<0.22,0.8,0.176>,
<0.24,0.8,0.192>,
<0.26,0.8,0.208>,
<0.28,0.8,0.224>,
<0.3,0.8,0.24>,
<0.32,0.8,0.256>,
<0.34,0.8,0.272>,
<0.36,0.8,0.288>,
<0.38,0.8,0.304>,
<0.4,0.8,0.32>,
<0.42,0.8,0.336>,
<0.44,0.8,0.352>,
<0.46,0.8,0.368>,
<0.48,0.8,0.384>,
<0.5,0.8,0.4>,
<0.52,0.8,0.416>,
<0.54,0.8,0.432>,
<0.56,0.8,0.448>,
<0.58,0.8,0.464>,
<0.6,0.8,0.48>,
<0.62,0.8,0.496>,
<0.64,0.8,0.512>,
<0.66,0.8,0.528>,
<0.68,0.8,0.544>,
<0.7,0.8,0.56>,

<0.72,0.8,0.576>,
<0.74,0.8,0.592>,
<0.76,0.8,0.608>,
<0.78,0.8,0.624>,
<0.8,0.8,0.64>,
<0.82,0.8,0.656>,
<0.84,0.8,0.672>,
<0.86,0.8,0.688>,
<0.88,0.8,0.704>,
<0.9,0.8,0.72>,
<0.92,0.8,0.736>,
<0.94,0.8,0.752>,
<0.96,0.8,0.768>,
<0.98,0.8,0.784>,
<1,0.8,0.8>,
<0,0.82,0>,
<0.02,0.82,0.0164>,
<0.04,0.82,0.0328>,
<0.06,0.82,0.0492>,
<0.08,0.82,0.0656>,
<0.1,0.82,0.082>,
<0.12,0.82,0.0984>,
<0.14,0.82,0.1148>,
<0.16,0.82,0.1312>,
<0.18,0.82,0.1476>,
<0.2,0.82,0.164>,
<0.22,0.82,0.1804>,
<0.24,0.82,0.1968>,
<0.26,0.82,0.2132>,
<0.28,0.82,0.2296>,
<0.3,0.82,0.246>,
<0.32,0.82,0.2624>,
<0.34,0.82,0.2788>,
<0.36,0.82,0.2952>,
<0.38,0.82,0.3116>,
<0.4,0.82,0.328>,
<0.42,0.82,0.3444>,
<0.44,0.82,0.3608>,
<0.46,0.82,0.3772>,
<0.48,0.82,0.3936>,
<0.5,0.82,0.41>,
<0.52,0.82,0.4264>,
<0.54,0.82,0.4428>,
<0.56,0.82,0.4592>,
<0.58,0.82,0.4756>,
<0.6,0.82,0.492>,
<0.62,0.82,0.5084>,
<0.64,0.82,0.5248>,
<0.66,0.82,0.5412>,
<0.68,0.82,0.5576>,
<0.7,0.82,0.574>,
<0.72,0.82,0.5904>,
<0.74,0.82,0.6068>,
<0.76,0.82,0.6232>,
<0.78,0.82,0.6396>,
<0.8,0.82,0.656>,

<0.82,0.82,0.6724>,
<0.84,0.82,0.6888>,
<0.86,0.82,0.7052>,
<0.88,0.82,0.7216>,
<0.9,0.82,0.738>,
<0.92,0.82,0.7544>,
<0.94,0.82,0.7708>,
<0.96,0.82,0.7872>,
<0.98,0.82,0.8036>,
<1,0.82,0.82>,
<0,0.84,0>,
<0.02,0.84,0.0168>,
<0.04,0.84,0.0336>,
<0.06,0.84,0.0504>,
<0.08,0.84,0.0672>,
<0.1,0.84,0.084>,
<0.12,0.84,0.1008>,
<0.14,0.84,0.1176>,
<0.16,0.84,0.1344>,
<0.18,0.84,0.1512>,
<0.2,0.84,0.168>,
<0.22,0.84,0.1848>,
<0.24,0.84,0.2016>,
<0.26,0.84,0.2184>,
<0.28,0.84,0.2352>,
<0.3,0.84,0.252>,
<0.32,0.84,0.2688>,
<0.34,0.84,0.2856>,
<0.36,0.84,0.3024>,
<0.38,0.84,0.3192>,
<0.4,0.84,0.336>,
<0.42,0.84,0.3528>,
<0.44,0.84,0.3696>,
<0.46,0.84,0.3864>,
<0.48,0.84,0.4032>,
<0.5,0.84,0.42>,
<0.52,0.84,0.4368>,
<0.54,0.84,0.4536>,
<0.56,0.84,0.4704>,
<0.58,0.84,0.4872>,
<0.6,0.84,0.504>,
<0.62,0.84,0.5208>,
<0.64,0.84,0.5376>,
<0.66,0.84,0.5544>,
<0.68,0.84,0.5712>,
<0.7,0.84,0.588>,
<0.72,0.84,0.6048>,
<0.74,0.84,0.6216>,
<0.76,0.84,0.6384>,
<0.78,0.84,0.6552>,
<0.8,0.84,0.672>,
<0.82,0.84,0.6888>,
<0.84,0.84,0.7056>,
<0.86,0.84,0.7224>,
<0.88,0.84,0.7392>,
<0.9,0.84,0.756>,

<0.92,0.84,0.7728>,
<0.94,0.84,0.7896>,
<0.96,0.84,0.8064>,
<0.98,0.84,0.8232>,
<1,0.84,0.84>,
<0,0.86,0>,
<0.02,0.86,0.0172>,
<0.04,0.86,0.0344>,
<0.06,0.86,0.0516>,
<0.08,0.86,0.0688>,
<0.1,0.86,0.086>,
<0.12,0.86,0.1032>,
<0.14,0.86,0.1204>,
<0.16,0.86,0.1376>,
<0.18,0.86,0.1548>,
<0.2,0.86,0.172>,
<0.22,0.86,0.1892>,
<0.24,0.86,0.2064>,
<0.26,0.86,0.2236>,
<0.28,0.86,0.2408>,
<0.3,0.86,0.258>,
<0.32,0.86,0.2752>,
<0.34,0.86,0.2924>,
<0.36,0.86,0.3096>,
<0.38,0.86,0.3268>,
<0.4,0.86,0.344>,
<0.42,0.86,0.3612>,
<0.44,0.86,0.3784>,
<0.46,0.86,0.3956>,
<0.48,0.86,0.4128>,
<0.5,0.86,0.43>,
<0.52,0.86,0.4472>,
<0.54,0.86,0.4644>,
<0.56,0.86,0.4816>,
<0.58,0.86,0.4988>,
<0.6,0.86,0.516>,
<0.62,0.86,0.5332>,
<0.64,0.86,0.5504>,
<0.66,0.86,0.5676>,
<0.68,0.86,0.5848>,
<0.7,0.86,0.602>,
<0.72,0.86,0.6192>,
<0.74,0.86,0.6364>,
<0.76,0.86,0.6536>,
<0.78,0.86,0.6708>,
<0.8,0.86,0.688>,
<0.82,0.86,0.7052>,
<0.84,0.86,0.7224>,
<0.86,0.86,0.7396>,
<0.88,0.86,0.7568>,
<0.9,0.86,0.774>,
<0.92,0.86,0.7912>,
<0.94,0.86,0.8084>,
<0.96,0.86,0.8256>,
<0.98,0.86,0.8428>,
<1,0.86,0.86>,

<0,0.88,0>,
<0.02,0.88,0.0176>,
<0.04,0.88,0.0352>,
<0.06,0.88,0.0528>,
<0.08,0.88,0.0704>,
<0.1,0.88,0.088>,
<0.12,0.88,0.1056>,
<0.14,0.88,0.1232>,
<0.16,0.88,0.1408>,
<0.18,0.88,0.1584>,
<0.2,0.88,0.176>,
<0.22,0.88,0.1936>,
<0.24,0.88,0.2112>,
<0.26,0.88,0.2288>,
<0.28,0.88,0.2464>,
<0.3,0.88,0.264>,
<0.32,0.88,0.2816>,
<0.34,0.88,0.2992>,
<0.36,0.88,0.3168>,
<0.38,0.88,0.3344>,
<0.4,0.88,0.352>,
<0.42,0.88,0.3696>,
<0.44,0.88,0.3872>,
<0.46,0.88,0.4048>,
<0.48,0.88,0.4224>,
<0.5,0.88,0.44>,
<0.52,0.88,0.4576>,
<0.54,0.88,0.4752>,
<0.56,0.88,0.4928>,
<0.58,0.88,0.5104>,
<0.6,0.88,0.528>,
<0.62,0.88,0.5456>,
<0.64,0.88,0.5632>,
<0.66,0.88,0.5808>,
<0.68,0.88,0.5984>,
<0.7,0.88,0.616>,
<0.72,0.88,0.6336>,
<0.74,0.88,0.6512>,
<0.76,0.88,0.6688>,
<0.78,0.88,0.6864>,
<0.8,0.88,0.704>,
<0.82,0.88,0.7216>,
<0.84,0.88,0.7392>,
<0.86,0.88,0.7568>,
<0.88,0.88,0.7744>,
<0.9,0.88,0.792>,
<0.92,0.88,0.8096>,
<0.94,0.88,0.8272>,
<0.96,0.88,0.8448>,
<0.98,0.88,0.8624>,
<1,0.88,0.88>,
<0,0.9,0>,
<0.02,0.9,0.018>,
<0.04,0.9,0.036>,
<0.06,0.9,0.054>,
<0.08,0.9,0.072>,

<0.1,0.9,0.09>,
<0.12,0.9,0.108>,
<0.14,0.9,0.126>,
<0.16,0.9,0.144>,
<0.18,0.9,0.162>,
<0.2,0.9,0.18>,
<0.22,0.9,0.198>,
<0.24,0.9,0.216>,
<0.26,0.9,0.234>,
<0.28,0.9,0.252>,
<0.3,0.9,0.27>,
<0.32,0.9,0.288>,
<0.34,0.9,0.306>,
<0.36,0.9,0.324>,
<0.38,0.9,0.342>,
<0.4,0.9,0.36>,
<0.42,0.9,0.378>,
<0.44,0.9,0.396>,
<0.46,0.9,0.414>,
<0.48,0.9,0.432>,
<0.5,0.9,0.45>,
<0.52,0.9,0.468>,
<0.54,0.9,0.486>,
<0.56,0.9,0.504>,
<0.58,0.9,0.522>,
<0.6,0.9,0.54>,
<0.62,0.9,0.558>,
<0.64,0.9,0.576>,
<0.66,0.9,0.594>,
<0.68,0.9,0.612>,
<0.7,0.9,0.63>,
<0.72,0.9,0.648>,
<0.74,0.9,0.666>,
<0.76,0.9,0.684>,
<0.78,0.9,0.702>,
<0.8,0.9,0.72>,
<0.82,0.9,0.738>,
<0.84,0.9,0.756>,
<0.86,0.9,0.774>,
<0.88,0.9,0.792>,
<0.9,0.9,0.81>,
<0.92,0.9,0.828>,
<0.94,0.9,0.846>,
<0.96,0.9,0.864>,
<0.98,0.9,0.882>,
<1,0.9,0.9>,
<0,0.92,0>,
<0.02,0.92,0.0184>,
<0.04,0.92,0.0368>,
<0.06,0.92,0.0552>,
<0.08,0.92,0.0736>,
<0.1,0.92,0.092>,
<0.12,0.92,0.1104>,
<0.14,0.92,0.1288>,
<0.16,0.92,0.1472>,
<0.18,0.92,0.1656>,

<0.2,0.92,0.184>,
<0.22,0.92,0.2024>,
<0.24,0.92,0.2208>,
<0.26,0.92,0.2392>,
<0.28,0.92,0.2576>,
<0.3,0.92,0.276>,
<0.32,0.92,0.2944>,
<0.34,0.92,0.3128>,
<0.36,0.92,0.3312>,
<0.38,0.92,0.3496>,
<0.4,0.92,0.368>,
<0.42,0.92,0.3864>,
<0.44,0.92,0.4048>,
<0.46,0.92,0.4232>,
<0.48,0.92,0.4416>,
<0.5,0.92,0.46>,
<0.52,0.92,0.4784>,
<0.54,0.92,0.4968>,
<0.56,0.92,0.5152>,
<0.58,0.92,0.5336>,
<0.6,0.92,0.552>,
<0.62,0.92,0.5704>,
<0.64,0.92,0.5888>,
<0.66,0.92,0.6072>,
<0.68,0.92,0.6256>,
<0.7,0.92,0.644>,
<0.72,0.92,0.6624>,
<0.74,0.92,0.6808>,
<0.76,0.92,0.6992>,
<0.78,0.92,0.7176>,
<0.8,0.92,0.736>,
<0.82,0.92,0.7544>,
<0.84,0.92,0.7728>,
<0.86,0.92,0.7912>,
<0.88,0.92,0.8096>,
<0.9,0.92,0.828>,
<0.92,0.92,0.8464>,
<0.94,0.92,0.8648>,
<0.96,0.92,0.8832>,
<0.98,0.92,0.9016>,
<1,0.92,0.92>,
<0,0.94,0>,
<0.02,0.94,0.0188>,
<0.04,0.94,0.0376>,
<0.06,0.94,0.0564>,
<0.08,0.94,0.0752>,
<0.1,0.94,0.094>,
<0.12,0.94,0.1128>,
<0.14,0.94,0.1316>,
<0.16,0.94,0.1504>,
<0.18,0.94,0.1692>,
<0.2,0.94,0.188>,
<0.22,0.94,0.2068>,
<0.24,0.94,0.2256>,
<0.26,0.94,0.2444>,
<0.28,0.94,0.2632>,

<0.3,0.94,0.282>,
<0.32,0.94,0.3008>,
<0.34,0.94,0.3196>,
<0.36,0.94,0.3384>,
<0.38,0.94,0.3572>,
<0.4,0.94,0.376>,
<0.42,0.94,0.3948>,
<0.44,0.94,0.4136>,
<0.46,0.94,0.4324>,
<0.48,0.94,0.4512>,
<0.5,0.94,0.47>,
<0.52,0.94,0.4888>,
<0.54,0.94,0.5076>,
<0.56,0.94,0.5264>,
<0.58,0.94,0.5452>,
<0.6,0.94,0.564>,
<0.62,0.94,0.5828>,
<0.64,0.94,0.6016>,
<0.66,0.94,0.6204>,
<0.68,0.94,0.6392>,
<0.7,0.94,0.658>,
<0.72,0.94,0.6768>,
<0.74,0.94,0.6956>,
<0.76,0.94,0.7144>,
<0.78,0.94,0.7332>,
<0.8,0.94,0.752>,
<0.82,0.94,0.7708>,
<0.84,0.94,0.7896>,
<0.86,0.94,0.8084>,
<0.88,0.94,0.8272>,
<0.9,0.94,0.846>,
<0.92,0.94,0.8648>,
<0.94,0.94,0.8836>,
<0.96,0.94,0.9024>,
<0.98,0.94,0.9212>,
<1,0.94,0.94>,
<0,0.96,0>,
<0.02,0.96,0.0192>,
<0.04,0.96,0.0384>,
<0.06,0.96,0.0576>,
<0.08,0.96,0.0768>,
<0.1,0.96,0.096>,
<0.12,0.96,0.1152>,
<0.14,0.96,0.1344>,
<0.16,0.96,0.1536>,
<0.18,0.96,0.1728>,
<0.2,0.96,0.192>,
<0.22,0.96,0.2112>,
<0.24,0.96,0.2304>,
<0.26,0.96,0.2496>,
<0.28,0.96,0.2688>,
<0.3,0.96,0.288>,
<0.32,0.96,0.3072>,
<0.34,0.96,0.3264>,
<0.36,0.96,0.3456>,
<0.38,0.96,0.3648>,

<0.4,0.96,0.384>,
<0.42,0.96,0.4032>,
<0.44,0.96,0.4224>,
<0.46,0.96,0.4416>,
<0.48,0.96,0.4608>,
<0.5,0.96,0.48>,
<0.52,0.96,0.4992>,
<0.54,0.96,0.5184>,
<0.56,0.96,0.5376>,
<0.58,0.96,0.5568>,
<0.6,0.96,0.576>,
<0.62,0.96,0.5952>,
<0.64,0.96,0.6144>,
<0.66,0.96,0.6336>,
<0.68,0.96,0.6528>,
<0.7,0.96,0.672>,
<0.72,0.96,0.6912>,
<0.74,0.96,0.7104>,
<0.76,0.96,0.7296>,
<0.78,0.96,0.7488>,
<0.8,0.96,0.768>,
<0.82,0.96,0.7872>,
<0.84,0.96,0.8064>,
<0.86,0.96,0.8256>,
<0.88,0.96,0.8448>,
<0.9,0.96,0.864>,
<0.92,0.96,0.8832>,
<0.94,0.96,0.9024>,
<0.96,0.96,0.9216>,
<0.98,0.96,0.9408>,
<1,0.96,0.96>,
<0,0.98,0>,
<0.02,0.98,0.0196>,
<0.04,0.98,0.0392>,
<0.06,0.98,0.0588>,
<0.08,0.98,0.0784>,
<0.1,0.98,0.098>,
<0.12,0.98,0.1176>,
<0.14,0.98,0.1372>,
<0.16,0.98,0.1568>,
<0.18,0.98,0.1764>,
<0.2,0.98,0.196>,
<0.22,0.98,0.2156>,
<0.24,0.98,0.2352>,
<0.26,0.98,0.2548>,
<0.28,0.98,0.2744>,
<0.3,0.98,0.294>,
<0.32,0.98,0.3136>,
<0.34,0.98,0.3332>,
<0.36,0.98,0.3528>,
<0.38,0.98,0.3724>,
<0.4,0.98,0.392>,
<0.42,0.98,0.4116>,
<0.44,0.98,0.4312>,
<0.46,0.98,0.4508>,
<0.48,0.98,0.4704>,

<0.5,0.98,0.49>,
<0.52,0.98,0.5096>,
<0.54,0.98,0.5292>,
<0.56,0.98,0.5488>,
<0.58,0.98,0.5684>,
<0.6,0.98,0.588>,
<0.62,0.98,0.6076>,
<0.64,0.98,0.6272>,
<0.66,0.98,0.6468>,
<0.68,0.98,0.6664>,
<0.7,0.98,0.686>,
<0.72,0.98,0.7056>,
<0.74,0.98,0.7252>,
<0.76,0.98,0.7448>,
<0.78,0.98,0.7644>,
<0.8,0.98,0.784>,
<0.82,0.98,0.8036>,
<0.84,0.98,0.8232>,
<0.86,0.98,0.8428>,
<0.88,0.98,0.8624>,
<0.9,0.98,0.882>,
<0.92,0.98,0.9016>,
<0.94,0.98,0.9212>,
<0.96,0.98,0.9408>,
<0.98,0.98,0.9604>,
<1,0.98,0.98>,
<0,1,0>,
<0.02,1,0.02>,
<0.04,1,0.04>,
<0.06,1,0.06>,
<0.08,1,0.08>,
<0.1,1,0.1>,
<0.12,1,0.12>,
<0.14,1,0.14>,
<0.16,1,0.16>,
<0.18,1,0.18>,
<0.2,1,0.2>,
<0.22,1,0.22>,
<0.24,1,0.24>,
<0.26,1,0.26>,
<0.28,1,0.28>,
<0.3,1,0.3>,
<0.32,1,0.32>,
<0.34,1,0.34>,
<0.36,1,0.36>,
<0.38,1,0.38>,
<0.4,1,0.4>,
<0.42,1,0.42>,
<0.44,1,0.44>,
<0.46,1,0.46>,
<0.48,1,0.48>,
<0.5,1,0.5>,
<0.52,1,0.52>,
<0.54,1,0.54>,
<0.56,1,0.56>,
<0.58,1,0.58>,

```

<0.6,1,0.6>,
<0.62,1,0.62>,
<0.64,1,0.64>,
<0.66,1,0.66>,
<0.68,1,0.68>,
<0.7,1,0.7>,
<0.72,1,0.72>,
<0.74,1,0.74>,
<0.76,1,0.76>,
<0.78,1,0.78>,
<0.8,1,0.8>,
<0.82,1,0.82>,
<0.84,1,0.84>,
<0.86,1,0.86>,
<0.88,1,0.88>,
<0.9,1,0.9>,
<0.92,1,0.92>,
<0.94,1,0.94>,
<0.96,1,0.96>,
<0.98,1,0.98>,
<1,1,1>,
}
normal_vectors { 2601,
<0,0,1>,
<0,-0.02,1>,
<0,-0.04,1>,
<0,-0.06,1>,
<0,-0.08,1>,
<0,-0.1,1>,
<0,-0.12,1>,
<0,-0.14,1>,
<0,-0.16,1>,
<0,-0.18,1>,
<0,-0.2,1>,
<0,-0.22,1>,
<0,-0.24,1>,
<0,-0.26,1>,
<0,-0.28,1>,
<0,-0.3,1>,
<0,-0.32,1>,
<0,-0.34,1>,
<0,-0.36,1>,
<0,-0.38,1>,
<0,-0.4,1>,
<0,-0.42,1>,
<0,-0.44,1>,
<0,-0.46,1>,
<0,-0.48,1>,
<0,-0.5,1>,
<0,-0.52,1>,
<0,-0.54,1>,
<0,-0.56,1>,
<0,-0.58,1>,
<0,-0.6,1>,
<0,-0.62,1>,
<0,-0.64,1>,

```

<0,-0.66,1>,
<0,-0.68,1>,
<0,-0.7,1>,
<0,-0.72,1>,
<0,-0.74,1>,
<0,-0.76,1>,
<0,-0.78,1>,
<0,-0.8,1>,
<0,-0.82,1>,
<0,-0.84,1>,
<0,-0.86,1>,
<0,-0.88,1>,
<0,-0.9,1>,
<0,-0.92,1>,
<0,-0.94,1>,
<0,-0.96,1>,
<0,-0.98,1>,
<0,-1,1>,
<-0.02,0,1>,
<-0.02,-0.02,1>,
<-0.02,-0.04,1>,
<-0.02,-0.06,1>,
<-0.02,-0.08,1>,
<-0.02,-0.1,1>,
<-0.02,-0.12,1>,
<-0.02,-0.14,1>,
<-0.02,-0.16,1>,
<-0.02,-0.18,1>,
<-0.02,-0.2,1>,
<-0.02,-0.22,1>,
<-0.02,-0.24,1>,
<-0.02,-0.26,1>,
<-0.02,-0.28,1>,
<-0.02,-0.3,1>,
<-0.02,-0.32,1>,
<-0.02,-0.34,1>,
<-0.02,-0.36,1>,
<-0.02,-0.38,1>,
<-0.02,-0.4,1>,
<-0.02,-0.42,1>,
<-0.02,-0.44,1>,
<-0.02,-0.46,1>,
<-0.02,-0.48,1>,
<-0.02,-0.5,1>,
<-0.02,-0.52,1>,
<-0.02,-0.54,1>,
<-0.02,-0.56,1>,
<-0.02,-0.58,1>,
<-0.02,-0.6,1>,
<-0.02,-0.62,1>,
<-0.02,-0.64,1>,
<-0.02,-0.66,1>,
<-0.02,-0.68,1>,
<-0.02,-0.7,1>,
<-0.02,-0.72,1>,
<-0.02,-0.74,1>,

<-0.02,-0.76,1>,
<-0.02,-0.78,1>,
<-0.02,-0.8,1>,
<-0.02,-0.82,1>,
<-0.02,-0.84,1>,
<-0.02,-0.86,1>,
<-0.02,-0.88,1>,
<-0.02,-0.9,1>,
<-0.02,-0.92,1>,
<-0.02,-0.94,1>,
<-0.02,-0.96,1>,
<-0.02,-0.98,1>,
<-0.02,-1,1>,
<-0.04,0,1>,
<-0.04,-0.02,1>,
<-0.04,-0.04,1>,
<-0.04,-0.06,1>,
<-0.04,-0.08,1>,
<-0.04,-0.1,1>,
<-0.04,-0.12,1>,
<-0.04,-0.14,1>,
<-0.04,-0.16,1>,
<-0.04,-0.18,1>,
<-0.04,-0.2,1>,
<-0.04,-0.22,1>,
<-0.04,-0.24,1>,
<-0.04,-0.26,1>,
<-0.04,-0.28,1>,
<-0.04,-0.3,1>,
<-0.04,-0.32,1>,
<-0.04,-0.34,1>,
<-0.04,-0.36,1>,
<-0.04,-0.38,1>,
<-0.04,-0.4,1>,
<-0.04,-0.42,1>,
<-0.04,-0.44,1>,
<-0.04,-0.46,1>,
<-0.04,-0.48,1>,
<-0.04,-0.5,1>,
<-0.04,-0.52,1>,
<-0.04,-0.54,1>,
<-0.04,-0.56,1>,
<-0.04,-0.58,1>,
<-0.04,-0.6,1>,
<-0.04,-0.62,1>,
<-0.04,-0.64,1>,
<-0.04,-0.66,1>,
<-0.04,-0.68,1>,
<-0.04,-0.7,1>,
<-0.04,-0.72,1>,
<-0.04,-0.74,1>,
<-0.04,-0.76,1>,
<-0.04,-0.78,1>,
<-0.04,-0.8,1>,
<-0.04,-0.82,1>,
<-0.04,-0.84,1>,

<-0.04,-0.86,1>,
<-0.04,-0.88,1>,
<-0.04,-0.9,1>,
<-0.04,-0.92,1>,
<-0.04,-0.94,1>,
<-0.04,-0.96,1>,
<-0.04,-0.98,1>,
<-0.04,-1,1>,
<-0.06,0,1>,
<-0.06,-0.02,1>,
<-0.06,-0.04,1>,
<-0.06,-0.06,1>,
<-0.06,-0.08,1>,
<-0.06,-0.1,1>,
<-0.06,-0.12,1>,
<-0.06,-0.14,1>,
<-0.06,-0.16,1>,
<-0.06,-0.18,1>,
<-0.06,-0.2,1>,
<-0.06,-0.22,1>,
<-0.06,-0.24,1>,
<-0.06,-0.26,1>,
<-0.06,-0.28,1>,
<-0.06,-0.3,1>,
<-0.06,-0.32,1>,
<-0.06,-0.34,1>,
<-0.06,-0.36,1>,
<-0.06,-0.38,1>,
<-0.06,-0.4,1>,
<-0.06,-0.42,1>,
<-0.06,-0.44,1>,
<-0.06,-0.46,1>,
<-0.06,-0.48,1>,
<-0.06,-0.5,1>,
<-0.06,-0.52,1>,
<-0.06,-0.54,1>,
<-0.06,-0.56,1>,
<-0.06,-0.58,1>,
<-0.06,-0.6,1>,
<-0.06,-0.62,1>,
<-0.06,-0.64,1>,
<-0.06,-0.66,1>,
<-0.06,-0.68,1>,
<-0.06,-0.7,1>,
<-0.06,-0.72,1>,
<-0.06,-0.74,1>,
<-0.06,-0.76,1>,
<-0.06,-0.78,1>,
<-0.06,-0.8,1>,
<-0.06,-0.82,1>,
<-0.06,-0.84,1>,
<-0.06,-0.86,1>,
<-0.06,-0.88,1>,
<-0.06,-0.9,1>,
<-0.06,-0.92,1>,
<-0.06,-0.94,1>,

<-0.06,-0.96,1>,
<-0.06,-0.98,1>,
<-0.06,-1,1>,
<-0.08,0,1>,
<-0.08,-0.02,1>,
<-0.08,-0.04,1>,
<-0.08,-0.06,1>,
<-0.08,-0.08,1>,
<-0.08,-0.1,1>,
<-0.08,-0.12,1>,
<-0.08,-0.14,1>,
<-0.08,-0.16,1>,
<-0.08,-0.18,1>,
<-0.08,-0.2,1>,
<-0.08,-0.22,1>,
<-0.08,-0.24,1>,
<-0.08,-0.26,1>,
<-0.08,-0.28,1>,
<-0.08,-0.3,1>,
<-0.08,-0.32,1>,
<-0.08,-0.34,1>,
<-0.08,-0.36,1>,
<-0.08,-0.38,1>,
<-0.08,-0.4,1>,
<-0.08,-0.42,1>,
<-0.08,-0.44,1>,
<-0.08,-0.46,1>,
<-0.08,-0.48,1>,
<-0.08,-0.5,1>,
<-0.08,-0.52,1>,
<-0.08,-0.54,1>,
<-0.08,-0.56,1>,
<-0.08,-0.58,1>,
<-0.08,-0.6,1>,
<-0.08,-0.62,1>,
<-0.08,-0.64,1>,
<-0.08,-0.66,1>,
<-0.08,-0.68,1>,
<-0.08,-0.7,1>,
<-0.08,-0.72,1>,
<-0.08,-0.74,1>,
<-0.08,-0.76,1>,
<-0.08,-0.78,1>,
<-0.08,-0.8,1>,
<-0.08,-0.82,1>,
<-0.08,-0.84,1>,
<-0.08,-0.86,1>,
<-0.08,-0.88,1>,
<-0.08,-0.9,1>,
<-0.08,-0.92,1>,
<-0.08,-0.94,1>,
<-0.08,-0.96,1>,
<-0.08,-0.98,1>,
<-0.08,-1,1>,
<-0.1,0,1>,
<-0.1,-0.02,1>,

<-0.1,-0.04,1>,
<-0.1,-0.06,1>,
<-0.1,-0.08,1>,
<-0.1,-0.1,1>,
<-0.1,-0.12,1>,
<-0.1,-0.14,1>,
<-0.1,-0.16,1>,
<-0.1,-0.18,1>,
<-0.1,-0.2,1>,
<-0.1,-0.22,1>,
<-0.1,-0.24,1>,
<-0.1,-0.26,1>,
<-0.1,-0.28,1>,
<-0.1,-0.3,1>,
<-0.1,-0.32,1>,
<-0.1,-0.34,1>,
<-0.1,-0.36,1>,
<-0.1,-0.38,1>,
<-0.1,-0.4,1>,
<-0.1,-0.42,1>,
<-0.1,-0.44,1>,
<-0.1,-0.46,1>,
<-0.1,-0.48,1>,
<-0.1,-0.5,1>,
<-0.1,-0.52,1>,
<-0.1,-0.54,1>,
<-0.1,-0.56,1>,
<-0.1,-0.58,1>,
<-0.1,-0.6,1>,
<-0.1,-0.62,1>,
<-0.1,-0.64,1>,
<-0.1,-0.66,1>,
<-0.1,-0.68,1>,
<-0.1,-0.7,1>,
<-0.1,-0.72,1>,
<-0.1,-0.74,1>,
<-0.1,-0.76,1>,
<-0.1,-0.78,1>,
<-0.1,-0.8,1>,
<-0.1,-0.82,1>,
<-0.1,-0.84,1>,
<-0.1,-0.86,1>,
<-0.1,-0.88,1>,
<-0.1,-0.9,1>,
<-0.1,-0.92,1>,
<-0.1,-0.94,1>,
<-0.1,-0.96,1>,
<-0.1,-0.98,1>,
<-0.1,-1,1>,
<-0.12,0,1>,
<-0.12,-0.02,1>,
<-0.12,-0.04,1>,
<-0.12,-0.06,1>,
<-0.12,-0.08,1>,
<-0.12,-0.1,1>,
<-0.12,-0.12,1>,

<-0.12,-0.14,1>,
<-0.12,-0.16,1>,
<-0.12,-0.18,1>,
<-0.12,-0.2,1>,
<-0.12,-0.22,1>,
<-0.12,-0.24,1>,
<-0.12,-0.26,1>,
<-0.12,-0.28,1>,
<-0.12,-0.3,1>,
<-0.12,-0.32,1>,
<-0.12,-0.34,1>,
<-0.12,-0.36,1>,
<-0.12,-0.38,1>,
<-0.12,-0.4,1>,
<-0.12,-0.42,1>,
<-0.12,-0.44,1>,
<-0.12,-0.46,1>,
<-0.12,-0.48,1>,
<-0.12,-0.5,1>,
<-0.12,-0.52,1>,
<-0.12,-0.54,1>,
<-0.12,-0.56,1>,
<-0.12,-0.58,1>,
<-0.12,-0.6,1>,
<-0.12,-0.62,1>,
<-0.12,-0.64,1>,
<-0.12,-0.66,1>,
<-0.12,-0.68,1>,
<-0.12,-0.7,1>,
<-0.12,-0.72,1>,
<-0.12,-0.74,1>,
<-0.12,-0.76,1>,
<-0.12,-0.78,1>,
<-0.12,-0.8,1>,
<-0.12,-0.82,1>,
<-0.12,-0.84,1>,
<-0.12,-0.86,1>,
<-0.12,-0.88,1>,
<-0.12,-0.9,1>,
<-0.12,-0.92,1>,
<-0.12,-0.94,1>,
<-0.12,-0.96,1>,
<-0.12,-0.98,1>,
<-0.12,-1,1>,
<-0.14,0,1>,
<-0.14,-0.02,1>,
<-0.14,-0.04,1>,
<-0.14,-0.06,1>,
<-0.14,-0.08,1>,
<-0.14,-0.1,1>,
<-0.14,-0.12,1>,
<-0.14,-0.14,1>,
<-0.14,-0.16,1>,
<-0.14,-0.18,1>,
<-0.14,-0.2,1>,
<-0.14,-0.22,1>,

<-0.14,-0.24,1>,
<-0.14,-0.26,1>,
<-0.14,-0.28,1>,
<-0.14,-0.3,1>,
<-0.14,-0.32,1>,
<-0.14,-0.34,1>,
<-0.14,-0.36,1>,
<-0.14,-0.38,1>,
<-0.14,-0.4,1>,
<-0.14,-0.42,1>,
<-0.14,-0.44,1>,
<-0.14,-0.46,1>,
<-0.14,-0.48,1>,
<-0.14,-0.5,1>,
<-0.14,-0.52,1>,
<-0.14,-0.54,1>,
<-0.14,-0.56,1>,
<-0.14,-0.58,1>,
<-0.14,-0.6,1>,
<-0.14,-0.62,1>,
<-0.14,-0.64,1>,
<-0.14,-0.66,1>,
<-0.14,-0.68,1>,
<-0.14,-0.7,1>,
<-0.14,-0.72,1>,
<-0.14,-0.74,1>,
<-0.14,-0.76,1>,
<-0.14,-0.78,1>,
<-0.14,-0.8,1>,
<-0.14,-0.82,1>,
<-0.14,-0.84,1>,
<-0.14,-0.86,1>,
<-0.14,-0.88,1>,
<-0.14,-0.9,1>,
<-0.14,-0.92,1>,
<-0.14,-0.94,1>,
<-0.14,-0.96,1>,
<-0.14,-0.98,1>,
<-0.14,-1,1>,
<-0.16,0,1>,
<-0.16,-0.02,1>,
<-0.16,-0.04,1>,
<-0.16,-0.06,1>,
<-0.16,-0.08,1>,
<-0.16,-0.1,1>,
<-0.16,-0.12,1>,
<-0.16,-0.14,1>,
<-0.16,-0.16,1>,
<-0.16,-0.18,1>,
<-0.16,-0.2,1>,
<-0.16,-0.22,1>,
<-0.16,-0.24,1>,
<-0.16,-0.26,1>,
<-0.16,-0.28,1>,
<-0.16,-0.3,1>,
<-0.16,-0.32,1>,

<-0.16,-0.34,1>,
<-0.16,-0.36,1>,
<-0.16,-0.38,1>,
<-0.16,-0.4,1>,
<-0.16,-0.42,1>,
<-0.16,-0.44,1>,
<-0.16,-0.46,1>,
<-0.16,-0.48,1>,
<-0.16,-0.5,1>,
<-0.16,-0.52,1>,
<-0.16,-0.54,1>,
<-0.16,-0.56,1>,
<-0.16,-0.58,1>,
<-0.16,-0.6,1>,
<-0.16,-0.62,1>,
<-0.16,-0.64,1>,
<-0.16,-0.66,1>,
<-0.16,-0.68,1>,
<-0.16,-0.7,1>,
<-0.16,-0.72,1>,
<-0.16,-0.74,1>,
<-0.16,-0.76,1>,
<-0.16,-0.78,1>,
<-0.16,-0.8,1>,
<-0.16,-0.82,1>,
<-0.16,-0.84,1>,
<-0.16,-0.86,1>,
<-0.16,-0.88,1>,
<-0.16,-0.9,1>,
<-0.16,-0.92,1>,
<-0.16,-0.94,1>,
<-0.16,-0.96,1>,
<-0.16,-0.98,1>,
<-0.16,-1,1>,
<-0.18,0,1>,
<-0.18,-0.02,1>,
<-0.18,-0.04,1>,
<-0.18,-0.06,1>,
<-0.18,-0.08,1>,
<-0.18,-0.1,1>,
<-0.18,-0.12,1>,
<-0.18,-0.14,1>,
<-0.18,-0.16,1>,
<-0.18,-0.18,1>,
<-0.18,-0.2,1>,
<-0.18,-0.22,1>,
<-0.18,-0.24,1>,
<-0.18,-0.26,1>,
<-0.18,-0.28,1>,
<-0.18,-0.3,1>,
<-0.18,-0.32,1>,
<-0.18,-0.34,1>,
<-0.18,-0.36,1>,
<-0.18,-0.38,1>,
<-0.18,-0.4,1>,
<-0.18,-0.42,1>,

<-0.18,-0.44,1>,
<-0.18,-0.46,1>,
<-0.18,-0.48,1>,
<-0.18,-0.5,1>,
<-0.18,-0.52,1>,
<-0.18,-0.54,1>,
<-0.18,-0.56,1>,
<-0.18,-0.58,1>,
<-0.18,-0.6,1>,
<-0.18,-0.62,1>,
<-0.18,-0.64,1>,
<-0.18,-0.66,1>,
<-0.18,-0.68,1>,
<-0.18,-0.7,1>,
<-0.18,-0.72,1>,
<-0.18,-0.74,1>,
<-0.18,-0.76,1>,
<-0.18,-0.78,1>,
<-0.18,-0.8,1>,
<-0.18,-0.82,1>,
<-0.18,-0.84,1>,
<-0.18,-0.86,1>,
<-0.18,-0.88,1>,
<-0.18,-0.9,1>,
<-0.18,-0.92,1>,
<-0.18,-0.94,1>,
<-0.18,-0.96,1>,
<-0.18,-0.98,1>,
<-0.18,-1,1>,
<-0.2,0,1>,
<-0.2,-0.02,1>,
<-0.2,-0.04,1>,
<-0.2,-0.06,1>,
<-0.2,-0.08,1>,
<-0.2,-0.1,1>,
<-0.2,-0.12,1>,
<-0.2,-0.14,1>,
<-0.2,-0.16,1>,
<-0.2,-0.18,1>,
<-0.2,-0.2,1>,
<-0.2,-0.22,1>,
<-0.2,-0.24,1>,
<-0.2,-0.26,1>,
<-0.2,-0.28,1>,
<-0.2,-0.3,1>,
<-0.2,-0.32,1>,
<-0.2,-0.34,1>,
<-0.2,-0.36,1>,
<-0.2,-0.38,1>,
<-0.2,-0.4,1>,
<-0.2,-0.42,1>,
<-0.2,-0.44,1>,
<-0.2,-0.46,1>,
<-0.2,-0.48,1>,
<-0.2,-0.5,1>,
<-0.2,-0.52,1>,

<-0.2,-0.54,1>,
<-0.2,-0.56,1>,
<-0.2,-0.58,1>,
<-0.2,-0.6,1>,
<-0.2,-0.62,1>,
<-0.2,-0.64,1>,
<-0.2,-0.66,1>,
<-0.2,-0.68,1>,
<-0.2,-0.7,1>,
<-0.2,-0.72,1>,
<-0.2,-0.74,1>,
<-0.2,-0.76,1>,
<-0.2,-0.78,1>,
<-0.2,-0.8,1>,
<-0.2,-0.82,1>,
<-0.2,-0.84,1>,
<-0.2,-0.86,1>,
<-0.2,-0.88,1>,
<-0.2,-0.9,1>,
<-0.2,-0.92,1>,
<-0.2,-0.94,1>,
<-0.2,-0.96,1>,
<-0.2,-0.98,1>,
<-0.2,-1,1>,
<-0.22,0,1>,
<-0.22,-0.02,1>,
<-0.22,-0.04,1>,
<-0.22,-0.06,1>,
<-0.22,-0.08,1>,
<-0.22,-0.1,1>,
<-0.22,-0.12,1>,
<-0.22,-0.14,1>,
<-0.22,-0.16,1>,
<-0.22,-0.18,1>,
<-0.22,-0.2,1>,
<-0.22,-0.22,1>,
<-0.22,-0.24,1>,
<-0.22,-0.26,1>,
<-0.22,-0.28,1>,
<-0.22,-0.3,1>,
<-0.22,-0.32,1>,
<-0.22,-0.34,1>,
<-0.22,-0.36,1>,
<-0.22,-0.38,1>,
<-0.22,-0.4,1>,
<-0.22,-0.42,1>,
<-0.22,-0.44,1>,
<-0.22,-0.46,1>,
<-0.22,-0.48,1>,
<-0.22,-0.5,1>,
<-0.22,-0.52,1>,
<-0.22,-0.54,1>,
<-0.22,-0.56,1>,
<-0.22,-0.58,1>,
<-0.22,-0.6,1>,
<-0.22,-0.62,1>,

<-0.22,-0.64,1>,
<-0.22,-0.66,1>,
<-0.22,-0.68,1>,
<-0.22,-0.7,1>,
<-0.22,-0.72,1>,
<-0.22,-0.74,1>,
<-0.22,-0.76,1>,
<-0.22,-0.78,1>,
<-0.22,-0.8,1>,
<-0.22,-0.82,1>,
<-0.22,-0.84,1>,
<-0.22,-0.86,1>,
<-0.22,-0.88,1>,
<-0.22,-0.9,1>,
<-0.22,-0.92,1>,
<-0.22,-0.94,1>,
<-0.22,-0.96,1>,
<-0.22,-0.98,1>,
<-0.22,-1,1>,
<-0.24,0,1>,
<-0.24,-0.02,1>,
<-0.24,-0.04,1>,
<-0.24,-0.06,1>,
<-0.24,-0.08,1>,
<-0.24,-0.1,1>,
<-0.24,-0.12,1>,
<-0.24,-0.14,1>,
<-0.24,-0.16,1>,
<-0.24,-0.18,1>,
<-0.24,-0.2,1>,
<-0.24,-0.22,1>,
<-0.24,-0.24,1>,
<-0.24,-0.26,1>,
<-0.24,-0.28,1>,
<-0.24,-0.3,1>,
<-0.24,-0.32,1>,
<-0.24,-0.34,1>,
<-0.24,-0.36,1>,
<-0.24,-0.38,1>,
<-0.24,-0.4,1>,
<-0.24,-0.42,1>,
<-0.24,-0.44,1>,
<-0.24,-0.46,1>,
<-0.24,-0.48,1>,
<-0.24,-0.5,1>,
<-0.24,-0.52,1>,
<-0.24,-0.54,1>,
<-0.24,-0.56,1>,
<-0.24,-0.58,1>,
<-0.24,-0.6,1>,
<-0.24,-0.62,1>,
<-0.24,-0.64,1>,
<-0.24,-0.66,1>,
<-0.24,-0.68,1>,
<-0.24,-0.7,1>,
<-0.24,-0.72,1>,

<-0.24,-0.74,1>,
<-0.24,-0.76,1>,
<-0.24,-0.78,1>,
<-0.24,-0.8,1>,
<-0.24,-0.82,1>,
<-0.24,-0.84,1>,
<-0.24,-0.86,1>,
<-0.24,-0.88,1>,
<-0.24,-0.9,1>,
<-0.24,-0.92,1>,
<-0.24,-0.94,1>,
<-0.24,-0.96,1>,
<-0.24,-0.98,1>,
<-0.24,-1,1>,
<-0.26,0,1>,
<-0.26,-0.02,1>,
<-0.26,-0.04,1>,
<-0.26,-0.06,1>,
<-0.26,-0.08,1>,
<-0.26,-0.1,1>,
<-0.26,-0.12,1>,
<-0.26,-0.14,1>,
<-0.26,-0.16,1>,
<-0.26,-0.18,1>,
<-0.26,-0.2,1>,
<-0.26,-0.22,1>,
<-0.26,-0.24,1>,
<-0.26,-0.26,1>,
<-0.26,-0.28,1>,
<-0.26,-0.3,1>,
<-0.26,-0.32,1>,
<-0.26,-0.34,1>,
<-0.26,-0.36,1>,
<-0.26,-0.38,1>,
<-0.26,-0.4,1>,
<-0.26,-0.42,1>,
<-0.26,-0.44,1>,
<-0.26,-0.46,1>,
<-0.26,-0.48,1>,
<-0.26,-0.5,1>,
<-0.26,-0.52,1>,
<-0.26,-0.54,1>,
<-0.26,-0.56,1>,
<-0.26,-0.58,1>,
<-0.26,-0.6,1>,
<-0.26,-0.62,1>,
<-0.26,-0.64,1>,
<-0.26,-0.66,1>,
<-0.26,-0.68,1>,
<-0.26,-0.7,1>,
<-0.26,-0.72,1>,
<-0.26,-0.74,1>,
<-0.26,-0.76,1>,
<-0.26,-0.78,1>,
<-0.26,-0.8,1>,
<-0.26,-0.82,1>,

<-0.26,-0.84,1>,
<-0.26,-0.86,1>,
<-0.26,-0.88,1>,
<-0.26,-0.9,1>,
<-0.26,-0.92,1>,
<-0.26,-0.94,1>,
<-0.26,-0.96,1>,
<-0.26,-0.98,1>,
<-0.26,-1,1>,
<-0.28,0,1>,
<-0.28,-0.02,1>,
<-0.28,-0.04,1>,
<-0.28,-0.06,1>,
<-0.28,-0.08,1>,
<-0.28,-0.1,1>,
<-0.28,-0.12,1>,
<-0.28,-0.14,1>,
<-0.28,-0.16,1>,
<-0.28,-0.18,1>,
<-0.28,-0.2,1>,
<-0.28,-0.22,1>,
<-0.28,-0.24,1>,
<-0.28,-0.26,1>,
<-0.28,-0.28,1>,
<-0.28,-0.3,1>,
<-0.28,-0.32,1>,
<-0.28,-0.34,1>,
<-0.28,-0.36,1>,
<-0.28,-0.38,1>,
<-0.28,-0.4,1>,
<-0.28,-0.42,1>,
<-0.28,-0.44,1>,
<-0.28,-0.46,1>,
<-0.28,-0.48,1>,
<-0.28,-0.5,1>,
<-0.28,-0.52,1>,
<-0.28,-0.54,1>,
<-0.28,-0.56,1>,
<-0.28,-0.58,1>,
<-0.28,-0.6,1>,
<-0.28,-0.62,1>,
<-0.28,-0.64,1>,
<-0.28,-0.66,1>,
<-0.28,-0.68,1>,
<-0.28,-0.7,1>,
<-0.28,-0.72,1>,
<-0.28,-0.74,1>,
<-0.28,-0.76,1>,
<-0.28,-0.78,1>,
<-0.28,-0.8,1>,
<-0.28,-0.82,1>,
<-0.28,-0.84,1>,
<-0.28,-0.86,1>,
<-0.28,-0.88,1>,
<-0.28,-0.9,1>,
<-0.28,-0.92,1>,

<-0.28,-0.94,1>,
<-0.28,-0.96,1>,
<-0.28,-0.98,1>,
<-0.28,-1,1>,
<-0.3,0,1>,
<-0.3,-0.02,1>,
<-0.3,-0.04,1>,
<-0.3,-0.06,1>,
<-0.3,-0.08,1>,
<-0.3,-0.1,1>,
<-0.3,-0.12,1>,
<-0.3,-0.14,1>,
<-0.3,-0.16,1>,
<-0.3,-0.18,1>,
<-0.3,-0.2,1>,
<-0.3,-0.22,1>,
<-0.3,-0.24,1>,
<-0.3,-0.26,1>,
<-0.3,-0.28,1>,
<-0.3,-0.3,1>,
<-0.3,-0.32,1>,
<-0.3,-0.34,1>,
<-0.3,-0.36,1>,
<-0.3,-0.38,1>,
<-0.3,-0.4,1>,
<-0.3,-0.42,1>,
<-0.3,-0.44,1>,
<-0.3,-0.46,1>,
<-0.3,-0.48,1>,
<-0.3,-0.5,1>,
<-0.3,-0.52,1>,
<-0.3,-0.54,1>,
<-0.3,-0.56,1>,
<-0.3,-0.58,1>,
<-0.3,-0.6,1>,
<-0.3,-0.62,1>,
<-0.3,-0.64,1>,
<-0.3,-0.66,1>,
<-0.3,-0.68,1>,
<-0.3,-0.7,1>,
<-0.3,-0.72,1>,
<-0.3,-0.74,1>,
<-0.3,-0.76,1>,
<-0.3,-0.78,1>,
<-0.3,-0.8,1>,
<-0.3,-0.82,1>,
<-0.3,-0.84,1>,
<-0.3,-0.86,1>,
<-0.3,-0.88,1>,
<-0.3,-0.9,1>,
<-0.3,-0.92,1>,
<-0.3,-0.94,1>,
<-0.3,-0.96,1>,
<-0.3,-0.98,1>,
<-0.3,-1,1>,
<-0.32,0,1>,

<-0.32,-0.02,1>,
<-0.32,-0.04,1>,
<-0.32,-0.06,1>,
<-0.32,-0.08,1>,
<-0.32,-0.1,1>,
<-0.32,-0.12,1>,
<-0.32,-0.14,1>,
<-0.32,-0.16,1>,
<-0.32,-0.18,1>,
<-0.32,-0.2,1>,
<-0.32,-0.22,1>,
<-0.32,-0.24,1>,
<-0.32,-0.26,1>,
<-0.32,-0.28,1>,
<-0.32,-0.3,1>,
<-0.32,-0.32,1>,
<-0.32,-0.34,1>,
<-0.32,-0.36,1>,
<-0.32,-0.38,1>,
<-0.32,-0.4,1>,
<-0.32,-0.42,1>,
<-0.32,-0.44,1>,
<-0.32,-0.46,1>,
<-0.32,-0.48,1>,
<-0.32,-0.5,1>,
<-0.32,-0.52,1>,
<-0.32,-0.54,1>,
<-0.32,-0.56,1>,
<-0.32,-0.58,1>,
<-0.32,-0.6,1>,
<-0.32,-0.62,1>,
<-0.32,-0.64,1>,
<-0.32,-0.66,1>,
<-0.32,-0.68,1>,
<-0.32,-0.7,1>,
<-0.32,-0.72,1>,
<-0.32,-0.74,1>,
<-0.32,-0.76,1>,
<-0.32,-0.78,1>,
<-0.32,-0.8,1>,
<-0.32,-0.82,1>,
<-0.32,-0.84,1>,
<-0.32,-0.86,1>,
<-0.32,-0.88,1>,
<-0.32,-0.9,1>,
<-0.32,-0.92,1>,
<-0.32,-0.94,1>,
<-0.32,-0.96,1>,
<-0.32,-0.98,1>,
<-0.32,-1,1>,
<-0.34,0,1>,
<-0.34,-0.02,1>,
<-0.34,-0.04,1>,
<-0.34,-0.06,1>,
<-0.34,-0.08,1>,
<-0.34,-0.1,1>,

<-0.34,-0.12,1>,
<-0.34,-0.14,1>,
<-0.34,-0.16,1>,
<-0.34,-0.18,1>,
<-0.34,-0.2,1>,
<-0.34,-0.22,1>,
<-0.34,-0.24,1>,
<-0.34,-0.26,1>,
<-0.34,-0.28,1>,
<-0.34,-0.3,1>,
<-0.34,-0.32,1>,
<-0.34,-0.34,1>,
<-0.34,-0.36,1>,
<-0.34,-0.38,1>,
<-0.34,-0.4,1>,
<-0.34,-0.42,1>,
<-0.34,-0.44,1>,
<-0.34,-0.46,1>,
<-0.34,-0.48,1>,
<-0.34,-0.5,1>,
<-0.34,-0.52,1>,
<-0.34,-0.54,1>,
<-0.34,-0.56,1>,
<-0.34,-0.58,1>,
<-0.34,-0.6,1>,
<-0.34,-0.62,1>,
<-0.34,-0.64,1>,
<-0.34,-0.66,1>,
<-0.34,-0.68,1>,
<-0.34,-0.7,1>,
<-0.34,-0.72,1>,
<-0.34,-0.74,1>,
<-0.34,-0.76,1>,
<-0.34,-0.78,1>,
<-0.34,-0.8,1>,
<-0.34,-0.82,1>,
<-0.34,-0.84,1>,
<-0.34,-0.86,1>,
<-0.34,-0.88,1>,
<-0.34,-0.9,1>,
<-0.34,-0.92,1>,
<-0.34,-0.94,1>,
<-0.34,-0.96,1>,
<-0.34,-0.98,1>,
<-0.34,-1,1>,
<-0.36,0,1>,
<-0.36,-0.02,1>,
<-0.36,-0.04,1>,
<-0.36,-0.06,1>,
<-0.36,-0.08,1>,
<-0.36,-0.1,1>,
<-0.36,-0.12,1>,
<-0.36,-0.14,1>,
<-0.36,-0.16,1>,
<-0.36,-0.18,1>,
<-0.36,-0.2,1>,

<-0.36,-0.22,1>,
<-0.36,-0.24,1>,
<-0.36,-0.26,1>,
<-0.36,-0.28,1>,
<-0.36,-0.3,1>,
<-0.36,-0.32,1>,
<-0.36,-0.34,1>,
<-0.36,-0.36,1>,
<-0.36,-0.38,1>,
<-0.36,-0.4,1>,
<-0.36,-0.42,1>,
<-0.36,-0.44,1>,
<-0.36,-0.46,1>,
<-0.36,-0.48,1>,
<-0.36,-0.5,1>,
<-0.36,-0.52,1>,
<-0.36,-0.54,1>,
<-0.36,-0.56,1>,
<-0.36,-0.58,1>,
<-0.36,-0.6,1>,
<-0.36,-0.62,1>,
<-0.36,-0.64,1>,
<-0.36,-0.66,1>,
<-0.36,-0.68,1>,
<-0.36,-0.7,1>,
<-0.36,-0.72,1>,
<-0.36,-0.74,1>,
<-0.36,-0.76,1>,
<-0.36,-0.78,1>,
<-0.36,-0.8,1>,
<-0.36,-0.82,1>,
<-0.36,-0.84,1>,
<-0.36,-0.86,1>,
<-0.36,-0.88,1>,
<-0.36,-0.9,1>,
<-0.36,-0.92,1>,
<-0.36,-0.94,1>,
<-0.36,-0.96,1>,
<-0.36,-0.98,1>,
<-0.36,-1,1>,
<-0.38,0,1>,
<-0.38,-0.02,1>,
<-0.38,-0.04,1>,
<-0.38,-0.06,1>,
<-0.38,-0.08,1>,
<-0.38,-0.1,1>,
<-0.38,-0.12,1>,
<-0.38,-0.14,1>,
<-0.38,-0.16,1>,
<-0.38,-0.18,1>,
<-0.38,-0.2,1>,
<-0.38,-0.22,1>,
<-0.38,-0.24,1>,
<-0.38,-0.26,1>,
<-0.38,-0.28,1>,
<-0.38,-0.3,1>,

<-0.38,-0.32,1>,
<-0.38,-0.34,1>,
<-0.38,-0.36,1>,
<-0.38,-0.38,1>,
<-0.38,-0.4,1>,
<-0.38,-0.42,1>,
<-0.38,-0.44,1>,
<-0.38,-0.46,1>,
<-0.38,-0.48,1>,
<-0.38,-0.5,1>,
<-0.38,-0.52,1>,
<-0.38,-0.54,1>,
<-0.38,-0.56,1>,
<-0.38,-0.58,1>,
<-0.38,-0.6,1>,
<-0.38,-0.62,1>,
<-0.38,-0.64,1>,
<-0.38,-0.66,1>,
<-0.38,-0.68,1>,
<-0.38,-0.7,1>,
<-0.38,-0.72,1>,
<-0.38,-0.74,1>,
<-0.38,-0.76,1>,
<-0.38,-0.78,1>,
<-0.38,-0.8,1>,
<-0.38,-0.82,1>,
<-0.38,-0.84,1>,
<-0.38,-0.86,1>,
<-0.38,-0.88,1>,
<-0.38,-0.9,1>,
<-0.38,-0.92,1>,
<-0.38,-0.94,1>,
<-0.38,-0.96,1>,
<-0.38,-0.98,1>,
<-0.38,-1,1>,
<-0.4,0,1>,
<-0.4,-0.02,1>,
<-0.4,-0.04,1>,
<-0.4,-0.06,1>,
<-0.4,-0.08,1>,
<-0.4,-0.1,1>,
<-0.4,-0.12,1>,
<-0.4,-0.14,1>,
<-0.4,-0.16,1>,
<-0.4,-0.18,1>,
<-0.4,-0.2,1>,
<-0.4,-0.22,1>,
<-0.4,-0.24,1>,
<-0.4,-0.26,1>,
<-0.4,-0.28,1>,
<-0.4,-0.3,1>,
<-0.4,-0.32,1>,
<-0.4,-0.34,1>,
<-0.4,-0.36,1>,
<-0.4,-0.38,1>,
<-0.4,-0.4,1>,

<-0.4,-0.42,1>,
<-0.4,-0.44,1>,
<-0.4,-0.46,1>,
<-0.4,-0.48,1>,
<-0.4,-0.5,1>,
<-0.4,-0.52,1>,
<-0.4,-0.54,1>,
<-0.4,-0.56,1>,
<-0.4,-0.58,1>,
<-0.4,-0.6,1>,
<-0.4,-0.62,1>,
<-0.4,-0.64,1>,
<-0.4,-0.66,1>,
<-0.4,-0.68,1>,
<-0.4,-0.7,1>,
<-0.4,-0.72,1>,
<-0.4,-0.74,1>,
<-0.4,-0.76,1>,
<-0.4,-0.78,1>,
<-0.4,-0.8,1>,
<-0.4,-0.82,1>,
<-0.4,-0.84,1>,
<-0.4,-0.86,1>,
<-0.4,-0.88,1>,
<-0.4,-0.9,1>,
<-0.4,-0.92,1>,
<-0.4,-0.94,1>,
<-0.4,-0.96,1>,
<-0.4,-0.98,1>,
<-0.4,-1,1>,
<-0.42,0,1>,
<-0.42,-0.02,1>,
<-0.42,-0.04,1>,
<-0.42,-0.06,1>,
<-0.42,-0.08,1>,
<-0.42,-0.1,1>,
<-0.42,-0.12,1>,
<-0.42,-0.14,1>,
<-0.42,-0.16,1>,
<-0.42,-0.18,1>,
<-0.42,-0.2,1>,
<-0.42,-0.22,1>,
<-0.42,-0.24,1>,
<-0.42,-0.26,1>,
<-0.42,-0.28,1>,
<-0.42,-0.3,1>,
<-0.42,-0.32,1>,
<-0.42,-0.34,1>,
<-0.42,-0.36,1>,
<-0.42,-0.38,1>,
<-0.42,-0.4,1>,
<-0.42,-0.42,1>,
<-0.42,-0.44,1>,
<-0.42,-0.46,1>,
<-0.42,-0.48,1>,
<-0.42,-0.5,1>,

<-0.42,-0.52,1>,
<-0.42,-0.54,1>,
<-0.42,-0.56,1>,
<-0.42,-0.58,1>,
<-0.42,-0.6,1>,
<-0.42,-0.62,1>,
<-0.42,-0.64,1>,
<-0.42,-0.66,1>,
<-0.42,-0.68,1>,
<-0.42,-0.7,1>,
<-0.42,-0.72,1>,
<-0.42,-0.74,1>,
<-0.42,-0.76,1>,
<-0.42,-0.78,1>,
<-0.42,-0.8,1>,
<-0.42,-0.82,1>,
<-0.42,-0.84,1>,
<-0.42,-0.86,1>,
<-0.42,-0.88,1>,
<-0.42,-0.9,1>,
<-0.42,-0.92,1>,
<-0.42,-0.94,1>,
<-0.42,-0.96,1>,
<-0.42,-0.98,1>,
<-0.42,-1,1>,
<-0.44,0,1>,
<-0.44,-0.02,1>,
<-0.44,-0.04,1>,
<-0.44,-0.06,1>,
<-0.44,-0.08,1>,
<-0.44,-0.1,1>,
<-0.44,-0.12,1>,
<-0.44,-0.14,1>,
<-0.44,-0.16,1>,
<-0.44,-0.18,1>,
<-0.44,-0.2,1>,
<-0.44,-0.22,1>,
<-0.44,-0.24,1>,
<-0.44,-0.26,1>,
<-0.44,-0.28,1>,
<-0.44,-0.3,1>,
<-0.44,-0.32,1>,
<-0.44,-0.34,1>,
<-0.44,-0.36,1>,
<-0.44,-0.38,1>,
<-0.44,-0.4,1>,
<-0.44,-0.42,1>,
<-0.44,-0.44,1>,
<-0.44,-0.46,1>,
<-0.44,-0.48,1>,
<-0.44,-0.5,1>,
<-0.44,-0.52,1>,
<-0.44,-0.54,1>,
<-0.44,-0.56,1>,
<-0.44,-0.58,1>,
<-0.44,-0.6,1>,

<-0.44,-0.62,1>,
<-0.44,-0.64,1>,
<-0.44,-0.66,1>,
<-0.44,-0.68,1>,
<-0.44,-0.7,1>,
<-0.44,-0.72,1>,
<-0.44,-0.74,1>,
<-0.44,-0.76,1>,
<-0.44,-0.78,1>,
<-0.44,-0.8,1>,
<-0.44,-0.82,1>,
<-0.44,-0.84,1>,
<-0.44,-0.86,1>,
<-0.44,-0.88,1>,
<-0.44,-0.9,1>,
<-0.44,-0.92,1>,
<-0.44,-0.94,1>,
<-0.44,-0.96,1>,
<-0.44,-0.98,1>,
<-0.44,-1,1>,
<-0.46,0,1>,
<-0.46,-0.02,1>,
<-0.46,-0.04,1>,
<-0.46,-0.06,1>,
<-0.46,-0.08,1>,
<-0.46,-0.1,1>,
<-0.46,-0.12,1>,
<-0.46,-0.14,1>,
<-0.46,-0.16,1>,
<-0.46,-0.18,1>,
<-0.46,-0.2,1>,
<-0.46,-0.22,1>,
<-0.46,-0.24,1>,
<-0.46,-0.26,1>,
<-0.46,-0.28,1>,
<-0.46,-0.3,1>,
<-0.46,-0.32,1>,
<-0.46,-0.34,1>,
<-0.46,-0.36,1>,
<-0.46,-0.38,1>,
<-0.46,-0.4,1>,
<-0.46,-0.42,1>,
<-0.46,-0.44,1>,
<-0.46,-0.46,1>,
<-0.46,-0.48,1>,
<-0.46,-0.5,1>,
<-0.46,-0.52,1>,
<-0.46,-0.54,1>,
<-0.46,-0.56,1>,
<-0.46,-0.58,1>,
<-0.46,-0.6,1>,
<-0.46,-0.62,1>,
<-0.46,-0.64,1>,
<-0.46,-0.66,1>,
<-0.46,-0.68,1>,
<-0.46,-0.7,1>,

<-0.46,-0.72,1>,
<-0.46,-0.74,1>,
<-0.46,-0.76,1>,
<-0.46,-0.78,1>,
<-0.46,-0.8,1>,
<-0.46,-0.82,1>,
<-0.46,-0.84,1>,
<-0.46,-0.86,1>,
<-0.46,-0.88,1>,
<-0.46,-0.9,1>,
<-0.46,-0.92,1>,
<-0.46,-0.94,1>,
<-0.46,-0.96,1>,
<-0.46,-0.98,1>,
<-0.46,-1,1>,
<-0.48,0,1>,
<-0.48,-0.02,1>,
<-0.48,-0.04,1>,
<-0.48,-0.06,1>,
<-0.48,-0.08,1>,
<-0.48,-0.1,1>,
<-0.48,-0.12,1>,
<-0.48,-0.14,1>,
<-0.48,-0.16,1>,
<-0.48,-0.18,1>,
<-0.48,-0.2,1>,
<-0.48,-0.22,1>,
<-0.48,-0.24,1>,
<-0.48,-0.26,1>,
<-0.48,-0.28,1>,
<-0.48,-0.3,1>,
<-0.48,-0.32,1>,
<-0.48,-0.34,1>,
<-0.48,-0.36,1>,
<-0.48,-0.38,1>,
<-0.48,-0.4,1>,
<-0.48,-0.42,1>,
<-0.48,-0.44,1>,
<-0.48,-0.46,1>,
<-0.48,-0.48,1>,
<-0.48,-0.5,1>,
<-0.48,-0.52,1>,
<-0.48,-0.54,1>,
<-0.48,-0.56,1>,
<-0.48,-0.58,1>,
<-0.48,-0.6,1>,
<-0.48,-0.62,1>,
<-0.48,-0.64,1>,
<-0.48,-0.66,1>,
<-0.48,-0.68,1>,
<-0.48,-0.7,1>,
<-0.48,-0.72,1>,
<-0.48,-0.74,1>,
<-0.48,-0.76,1>,
<-0.48,-0.78,1>,
<-0.48,-0.8,1>,

<-0.48,-0.82,1>,
<-0.48,-0.84,1>,
<-0.48,-0.86,1>,
<-0.48,-0.88,1>,
<-0.48,-0.9,1>,
<-0.48,-0.92,1>,
<-0.48,-0.94,1>,
<-0.48,-0.96,1>,
<-0.48,-0.98,1>,
<-0.48,-1,1>,
<-0.5,0,1>,
<-0.5,-0.02,1>,
<-0.5,-0.04,1>,
<-0.5,-0.06,1>,
<-0.5,-0.08,1>,
<-0.5,-0.1,1>,
<-0.5,-0.12,1>,
<-0.5,-0.14,1>,
<-0.5,-0.16,1>,
<-0.5,-0.18,1>,
<-0.5,-0.2,1>,
<-0.5,-0.22,1>,
<-0.5,-0.24,1>,
<-0.5,-0.26,1>,
<-0.5,-0.28,1>,
<-0.5,-0.3,1>,
<-0.5,-0.32,1>,
<-0.5,-0.34,1>,
<-0.5,-0.36,1>,
<-0.5,-0.38,1>,
<-0.5,-0.4,1>,
<-0.5,-0.42,1>,
<-0.5,-0.44,1>,
<-0.5,-0.46,1>,
<-0.5,-0.48,1>,
<-0.5,-0.5,1>,
<-0.5,-0.52,1>,
<-0.5,-0.54,1>,
<-0.5,-0.56,1>,
<-0.5,-0.58,1>,
<-0.5,-0.6,1>,
<-0.5,-0.62,1>,
<-0.5,-0.64,1>,
<-0.5,-0.66,1>,
<-0.5,-0.68,1>,
<-0.5,-0.7,1>,
<-0.5,-0.72,1>,
<-0.5,-0.74,1>,
<-0.5,-0.76,1>,
<-0.5,-0.78,1>,
<-0.5,-0.8,1>,
<-0.5,-0.82,1>,
<-0.5,-0.84,1>,
<-0.5,-0.86,1>,
<-0.5,-0.88,1>,
<-0.5,-0.9,1>,

<-0.5,-0.92,1>,
<-0.5,-0.94,1>,
<-0.5,-0.96,1>,
<-0.5,-0.98,1>,
<-0.5,-1,1>,
<-0.52,0,1>,
<-0.52,-0.02,1>,
<-0.52,-0.04,1>,
<-0.52,-0.06,1>,
<-0.52,-0.08,1>,
<-0.52,-0.1,1>,
<-0.52,-0.12,1>,
<-0.52,-0.14,1>,
<-0.52,-0.16,1>,
<-0.52,-0.18,1>,
<-0.52,-0.2,1>,
<-0.52,-0.22,1>,
<-0.52,-0.24,1>,
<-0.52,-0.26,1>,
<-0.52,-0.28,1>,
<-0.52,-0.3,1>,
<-0.52,-0.32,1>,
<-0.52,-0.34,1>,
<-0.52,-0.36,1>,
<-0.52,-0.38,1>,
<-0.52,-0.4,1>,
<-0.52,-0.42,1>,
<-0.52,-0.44,1>,
<-0.52,-0.46,1>,
<-0.52,-0.48,1>,
<-0.52,-0.5,1>,
<-0.52,-0.52,1>,
<-0.52,-0.54,1>,
<-0.52,-0.56,1>,
<-0.52,-0.58,1>,
<-0.52,-0.6,1>,
<-0.52,-0.62,1>,
<-0.52,-0.64,1>,
<-0.52,-0.66,1>,
<-0.52,-0.68,1>,
<-0.52,-0.7,1>,
<-0.52,-0.72,1>,
<-0.52,-0.74,1>,
<-0.52,-0.76,1>,
<-0.52,-0.78,1>,
<-0.52,-0.8,1>,
<-0.52,-0.82,1>,
<-0.52,-0.84,1>,
<-0.52,-0.86,1>,
<-0.52,-0.88,1>,
<-0.52,-0.9,1>,
<-0.52,-0.92,1>,
<-0.52,-0.94,1>,
<-0.52,-0.96,1>,
<-0.52,-0.98,1>,
<-0.52,-1,1>,

<-0.54, 0, 1>,
<-0.54, -0.02, 1>,
<-0.54, -0.04, 1>,
<-0.54, -0.06, 1>,
<-0.54, -0.08, 1>,
<-0.54, -0.1, 1>,
<-0.54, -0.12, 1>,
<-0.54, -0.14, 1>,
<-0.54, -0.16, 1>,
<-0.54, -0.18, 1>,
<-0.54, -0.2, 1>,
<-0.54, -0.22, 1>,
<-0.54, -0.24, 1>,
<-0.54, -0.26, 1>,
<-0.54, -0.28, 1>,
<-0.54, -0.3, 1>,
<-0.54, -0.32, 1>,
<-0.54, -0.34, 1>,
<-0.54, -0.36, 1>,
<-0.54, -0.38, 1>,
<-0.54, -0.4, 1>,
<-0.54, -0.42, 1>,
<-0.54, -0.44, 1>,
<-0.54, -0.46, 1>,
<-0.54, -0.48, 1>,
<-0.54, -0.5, 1>,
<-0.54, -0.52, 1>,
<-0.54, -0.54, 1>,
<-0.54, -0.56, 1>,
<-0.54, -0.58, 1>,
<-0.54, -0.6, 1>,
<-0.54, -0.62, 1>,
<-0.54, -0.64, 1>,
<-0.54, -0.66, 1>,
<-0.54, -0.68, 1>,
<-0.54, -0.7, 1>,
<-0.54, -0.72, 1>,
<-0.54, -0.74, 1>,
<-0.54, -0.76, 1>,
<-0.54, -0.78, 1>,
<-0.54, -0.8, 1>,
<-0.54, -0.82, 1>,
<-0.54, -0.84, 1>,
<-0.54, -0.86, 1>,
<-0.54, -0.88, 1>,
<-0.54, -0.9, 1>,
<-0.54, -0.92, 1>,
<-0.54, -0.94, 1>,
<-0.54, -0.96, 1>,
<-0.54, -0.98, 1>,
<-0.54, -1, 1>,
<-0.56, 0, 1>,
<-0.56, -0.02, 1>,
<-0.56, -0.04, 1>,
<-0.56, -0.06, 1>,
<-0.56, -0.08, 1>,

<-0.56,-0.1,1>,
<-0.56,-0.12,1>,
<-0.56,-0.14,1>,
<-0.56,-0.16,1>,
<-0.56,-0.18,1>,
<-0.56,-0.2,1>,
<-0.56,-0.22,1>,
<-0.56,-0.24,1>,
<-0.56,-0.26,1>,
<-0.56,-0.28,1>,
<-0.56,-0.3,1>,
<-0.56,-0.32,1>,
<-0.56,-0.34,1>,
<-0.56,-0.36,1>,
<-0.56,-0.38,1>,
<-0.56,-0.4,1>,
<-0.56,-0.42,1>,
<-0.56,-0.44,1>,
<-0.56,-0.46,1>,
<-0.56,-0.48,1>,
<-0.56,-0.5,1>,
<-0.56,-0.52,1>,
<-0.56,-0.54,1>,
<-0.56,-0.56,1>,
<-0.56,-0.58,1>,
<-0.56,-0.6,1>,
<-0.56,-0.62,1>,
<-0.56,-0.64,1>,
<-0.56,-0.66,1>,
<-0.56,-0.68,1>,
<-0.56,-0.7,1>,
<-0.56,-0.72,1>,
<-0.56,-0.74,1>,
<-0.56,-0.76,1>,
<-0.56,-0.78,1>,
<-0.56,-0.8,1>,
<-0.56,-0.82,1>,
<-0.56,-0.84,1>,
<-0.56,-0.86,1>,
<-0.56,-0.88,1>,
<-0.56,-0.9,1>,
<-0.56,-0.92,1>,
<-0.56,-0.94,1>,
<-0.56,-0.96,1>,
<-0.56,-0.98,1>,
<-0.56,-1,1>,
<-0.58,0,1>,
<-0.58,-0.02,1>,
<-0.58,-0.04,1>,
<-0.58,-0.06,1>,
<-0.58,-0.08,1>,
<-0.58,-0.1,1>,
<-0.58,-0.12,1>,
<-0.58,-0.14,1>,
<-0.58,-0.16,1>,
<-0.58,-0.18,1>,

<-0.58,-0.2,1>,
<-0.58,-0.22,1>,
<-0.58,-0.24,1>,
<-0.58,-0.26,1>,
<-0.58,-0.28,1>,
<-0.58,-0.3,1>,
<-0.58,-0.32,1>,
<-0.58,-0.34,1>,
<-0.58,-0.36,1>,
<-0.58,-0.38,1>,
<-0.58,-0.4,1>,
<-0.58,-0.42,1>,
<-0.58,-0.44,1>,
<-0.58,-0.46,1>,
<-0.58,-0.48,1>,
<-0.58,-0.5,1>,
<-0.58,-0.52,1>,
<-0.58,-0.54,1>,
<-0.58,-0.56,1>,
<-0.58,-0.58,1>,
<-0.58,-0.6,1>,
<-0.58,-0.62,1>,
<-0.58,-0.64,1>,
<-0.58,-0.66,1>,
<-0.58,-0.68,1>,
<-0.58,-0.7,1>,
<-0.58,-0.72,1>,
<-0.58,-0.74,1>,
<-0.58,-0.76,1>,
<-0.58,-0.78,1>,
<-0.58,-0.8,1>,
<-0.58,-0.82,1>,
<-0.58,-0.84,1>,
<-0.58,-0.86,1>,
<-0.58,-0.88,1>,
<-0.58,-0.9,1>,
<-0.58,-0.92,1>,
<-0.58,-0.94,1>,
<-0.58,-0.96,1>,
<-0.58,-0.98,1>,
<-0.58,-1,1>,
<-0.6,0,1>,
<-0.6,-0.02,1>,
<-0.6,-0.04,1>,
<-0.6,-0.06,1>,
<-0.6,-0.08,1>,
<-0.6,-0.1,1>,
<-0.6,-0.12,1>,
<-0.6,-0.14,1>,
<-0.6,-0.16,1>,
<-0.6,-0.18,1>,
<-0.6,-0.2,1>,
<-0.6,-0.22,1>,
<-0.6,-0.24,1>,
<-0.6,-0.26,1>,
<-0.6,-0.28,1>,

<-0.6,-0.3,1>,
<-0.6,-0.32,1>,
<-0.6,-0.34,1>,
<-0.6,-0.36,1>,
<-0.6,-0.38,1>,
<-0.6,-0.4,1>,
<-0.6,-0.42,1>,
<-0.6,-0.44,1>,
<-0.6,-0.46,1>,
<-0.6,-0.48,1>,
<-0.6,-0.5,1>,
<-0.6,-0.52,1>,
<-0.6,-0.54,1>,
<-0.6,-0.56,1>,
<-0.6,-0.58,1>,
<-0.6,-0.6,1>,
<-0.6,-0.62,1>,
<-0.6,-0.64,1>,
<-0.6,-0.66,1>,
<-0.6,-0.68,1>,
<-0.6,-0.7,1>,
<-0.6,-0.72,1>,
<-0.6,-0.74,1>,
<-0.6,-0.76,1>,
<-0.6,-0.78,1>,
<-0.6,-0.8,1>,
<-0.6,-0.82,1>,
<-0.6,-0.84,1>,
<-0.6,-0.86,1>,
<-0.6,-0.88,1>,
<-0.6,-0.9,1>,
<-0.6,-0.92,1>,
<-0.6,-0.94,1>,
<-0.6,-0.96,1>,
<-0.6,-0.98,1>,
<-0.6,-1,1>,
<-0.62,0,1>,
<-0.62,-0.02,1>,
<-0.62,-0.04,1>,
<-0.62,-0.06,1>,
<-0.62,-0.08,1>,
<-0.62,-0.1,1>,
<-0.62,-0.12,1>,
<-0.62,-0.14,1>,
<-0.62,-0.16,1>,
<-0.62,-0.18,1>,
<-0.62,-0.2,1>,
<-0.62,-0.22,1>,
<-0.62,-0.24,1>,
<-0.62,-0.26,1>,
<-0.62,-0.28,1>,
<-0.62,-0.3,1>,
<-0.62,-0.32,1>,
<-0.62,-0.34,1>,
<-0.62,-0.36,1>,
<-0.62,-0.38,1>,

<-0.62,-0.4,1>,
<-0.62,-0.42,1>,
<-0.62,-0.44,1>,
<-0.62,-0.46,1>,
<-0.62,-0.48,1>,
<-0.62,-0.5,1>,
<-0.62,-0.52,1>,
<-0.62,-0.54,1>,
<-0.62,-0.56,1>,
<-0.62,-0.58,1>,
<-0.62,-0.6,1>,
<-0.62,-0.62,1>,
<-0.62,-0.64,1>,
<-0.62,-0.66,1>,
<-0.62,-0.68,1>,
<-0.62,-0.7,1>,
<-0.62,-0.72,1>,
<-0.62,-0.74,1>,
<-0.62,-0.76,1>,
<-0.62,-0.78,1>,
<-0.62,-0.8,1>,
<-0.62,-0.82,1>,
<-0.62,-0.84,1>,
<-0.62,-0.86,1>,
<-0.62,-0.88,1>,
<-0.62,-0.9,1>,
<-0.62,-0.92,1>,
<-0.62,-0.94,1>,
<-0.62,-0.96,1>,
<-0.62,-0.98,1>,
<-0.62,-1,1>,
<-0.64,0,1>,
<-0.64,-0.02,1>,
<-0.64,-0.04,1>,
<-0.64,-0.06,1>,
<-0.64,-0.08,1>,
<-0.64,-0.1,1>,
<-0.64,-0.12,1>,
<-0.64,-0.14,1>,
<-0.64,-0.16,1>,
<-0.64,-0.18,1>,
<-0.64,-0.2,1>,
<-0.64,-0.22,1>,
<-0.64,-0.24,1>,
<-0.64,-0.26,1>,
<-0.64,-0.28,1>,
<-0.64,-0.3,1>,
<-0.64,-0.32,1>,
<-0.64,-0.34,1>,
<-0.64,-0.36,1>,
<-0.64,-0.38,1>,
<-0.64,-0.4,1>,
<-0.64,-0.42,1>,
<-0.64,-0.44,1>,
<-0.64,-0.46,1>,
<-0.64,-0.48,1>,

<-0.64,-0.5,1>,
<-0.64,-0.52,1>,
<-0.64,-0.54,1>,
<-0.64,-0.56,1>,
<-0.64,-0.58,1>,
<-0.64,-0.6,1>,
<-0.64,-0.62,1>,
<-0.64,-0.64,1>,
<-0.64,-0.66,1>,
<-0.64,-0.68,1>,
<-0.64,-0.7,1>,
<-0.64,-0.72,1>,
<-0.64,-0.74,1>,
<-0.64,-0.76,1>,
<-0.64,-0.78,1>,
<-0.64,-0.8,1>,
<-0.64,-0.82,1>,
<-0.64,-0.84,1>,
<-0.64,-0.86,1>,
<-0.64,-0.88,1>,
<-0.64,-0.9,1>,
<-0.64,-0.92,1>,
<-0.64,-0.94,1>,
<-0.64,-0.96,1>,
<-0.64,-0.98,1>,
<-0.64,-1,1>,
<-0.66,0,1>,
<-0.66,-0.02,1>,
<-0.66,-0.04,1>,
<-0.66,-0.06,1>,
<-0.66,-0.08,1>,
<-0.66,-0.1,1>,
<-0.66,-0.12,1>,
<-0.66,-0.14,1>,
<-0.66,-0.16,1>,
<-0.66,-0.18,1>,
<-0.66,-0.2,1>,
<-0.66,-0.22,1>,
<-0.66,-0.24,1>,
<-0.66,-0.26,1>,
<-0.66,-0.28,1>,
<-0.66,-0.3,1>,
<-0.66,-0.32,1>,
<-0.66,-0.34,1>,
<-0.66,-0.36,1>,
<-0.66,-0.38,1>,
<-0.66,-0.4,1>,
<-0.66,-0.42,1>,
<-0.66,-0.44,1>,
<-0.66,-0.46,1>,
<-0.66,-0.48,1>,
<-0.66,-0.5,1>,
<-0.66,-0.52,1>,
<-0.66,-0.54,1>,
<-0.66,-0.56,1>,
<-0.66,-0.58,1>,

<-0.66,-0.6,1>,
<-0.66,-0.62,1>,
<-0.66,-0.64,1>,
<-0.66,-0.66,1>,
<-0.66,-0.68,1>,
<-0.66,-0.7,1>,
<-0.66,-0.72,1>,
<-0.66,-0.74,1>,
<-0.66,-0.76,1>,
<-0.66,-0.78,1>,
<-0.66,-0.8,1>,
<-0.66,-0.82,1>,
<-0.66,-0.84,1>,
<-0.66,-0.86,1>,
<-0.66,-0.88,1>,
<-0.66,-0.9,1>,
<-0.66,-0.92,1>,
<-0.66,-0.94,1>,
<-0.66,-0.96,1>,
<-0.66,-0.98,1>,
<-0.66,-1,1>,
<-0.68,0,1>,
<-0.68,-0.02,1>,
<-0.68,-0.04,1>,
<-0.68,-0.06,1>,
<-0.68,-0.08,1>,
<-0.68,-0.1,1>,
<-0.68,-0.12,1>,
<-0.68,-0.14,1>,
<-0.68,-0.16,1>,
<-0.68,-0.18,1>,
<-0.68,-0.2,1>,
<-0.68,-0.22,1>,
<-0.68,-0.24,1>,
<-0.68,-0.26,1>,
<-0.68,-0.28,1>,
<-0.68,-0.3,1>,
<-0.68,-0.32,1>,
<-0.68,-0.34,1>,
<-0.68,-0.36,1>,
<-0.68,-0.38,1>,
<-0.68,-0.4,1>,
<-0.68,-0.42,1>,
<-0.68,-0.44,1>,
<-0.68,-0.46,1>,
<-0.68,-0.48,1>,
<-0.68,-0.5,1>,
<-0.68,-0.52,1>,
<-0.68,-0.54,1>,
<-0.68,-0.56,1>,
<-0.68,-0.58,1>,
<-0.68,-0.6,1>,
<-0.68,-0.62,1>,
<-0.68,-0.64,1>,
<-0.68,-0.66,1>,
<-0.68,-0.68,1>,

<-0.68,-0.7,1>,
<-0.68,-0.72,1>,
<-0.68,-0.74,1>,
<-0.68,-0.76,1>,
<-0.68,-0.78,1>,
<-0.68,-0.8,1>,
<-0.68,-0.82,1>,
<-0.68,-0.84,1>,
<-0.68,-0.86,1>,
<-0.68,-0.88,1>,
<-0.68,-0.9,1>,
<-0.68,-0.92,1>,
<-0.68,-0.94,1>,
<-0.68,-0.96,1>,
<-0.68,-0.98,1>,
<-0.68,-1,1>,
<-0.7,0,1>,
<-0.7,-0.02,1>,
<-0.7,-0.04,1>,
<-0.7,-0.06,1>,
<-0.7,-0.08,1>,
<-0.7,-0.1,1>,
<-0.7,-0.12,1>,
<-0.7,-0.14,1>,
<-0.7,-0.16,1>,
<-0.7,-0.18,1>,
<-0.7,-0.2,1>,
<-0.7,-0.22,1>,
<-0.7,-0.24,1>,
<-0.7,-0.26,1>,
<-0.7,-0.28,1>,
<-0.7,-0.3,1>,
<-0.7,-0.32,1>,
<-0.7,-0.34,1>,
<-0.7,-0.36,1>,
<-0.7,-0.38,1>,
<-0.7,-0.4,1>,
<-0.7,-0.42,1>,
<-0.7,-0.44,1>,
<-0.7,-0.46,1>,
<-0.7,-0.48,1>,
<-0.7,-0.5,1>,
<-0.7,-0.52,1>,
<-0.7,-0.54,1>,
<-0.7,-0.56,1>,
<-0.7,-0.58,1>,
<-0.7,-0.6,1>,
<-0.7,-0.62,1>,
<-0.7,-0.64,1>,
<-0.7,-0.66,1>,
<-0.7,-0.68,1>,
<-0.7,-0.7,1>,
<-0.7,-0.72,1>,
<-0.7,-0.74,1>,
<-0.7,-0.76,1>,
<-0.7,-0.78,1>,

<-0.7,-0.8,1>,
<-0.7,-0.82,1>,
<-0.7,-0.84,1>,
<-0.7,-0.86,1>,
<-0.7,-0.88,1>,
<-0.7,-0.9,1>,
<-0.7,-0.92,1>,
<-0.7,-0.94,1>,
<-0.7,-0.96,1>,
<-0.7,-0.98,1>,
<-0.7,-1,1>,
<-0.72,0,1>,
<-0.72,-0.02,1>,
<-0.72,-0.04,1>,
<-0.72,-0.06,1>,
<-0.72,-0.08,1>,
<-0.72,-0.1,1>,
<-0.72,-0.12,1>,
<-0.72,-0.14,1>,
<-0.72,-0.16,1>,
<-0.72,-0.18,1>,
<-0.72,-0.2,1>,
<-0.72,-0.22,1>,
<-0.72,-0.24,1>,
<-0.72,-0.26,1>,
<-0.72,-0.28,1>,
<-0.72,-0.3,1>,
<-0.72,-0.32,1>,
<-0.72,-0.34,1>,
<-0.72,-0.36,1>,
<-0.72,-0.38,1>,
<-0.72,-0.4,1>,
<-0.72,-0.42,1>,
<-0.72,-0.44,1>,
<-0.72,-0.46,1>,
<-0.72,-0.48,1>,
<-0.72,-0.5,1>,
<-0.72,-0.52,1>,
<-0.72,-0.54,1>,
<-0.72,-0.56,1>,
<-0.72,-0.58,1>,
<-0.72,-0.6,1>,
<-0.72,-0.62,1>,
<-0.72,-0.64,1>,
<-0.72,-0.66,1>,
<-0.72,-0.68,1>,
<-0.72,-0.7,1>,
<-0.72,-0.72,1>,
<-0.72,-0.74,1>,
<-0.72,-0.76,1>,
<-0.72,-0.78,1>,
<-0.72,-0.8,1>,
<-0.72,-0.82,1>,
<-0.72,-0.84,1>,
<-0.72,-0.86,1>,
<-0.72,-0.88,1>,

<-0.72,-0.9,1>,
<-0.72,-0.92,1>,
<-0.72,-0.94,1>,
<-0.72,-0.96,1>,
<-0.72,-0.98,1>,
<-0.72,-1,1>,
<-0.74,0,1>,
<-0.74,-0.02,1>,
<-0.74,-0.04,1>,
<-0.74,-0.06,1>,
<-0.74,-0.08,1>,
<-0.74,-0.1,1>,
<-0.74,-0.12,1>,
<-0.74,-0.14,1>,
<-0.74,-0.16,1>,
<-0.74,-0.18,1>,
<-0.74,-0.2,1>,
<-0.74,-0.22,1>,
<-0.74,-0.24,1>,
<-0.74,-0.26,1>,
<-0.74,-0.28,1>,
<-0.74,-0.3,1>,
<-0.74,-0.32,1>,
<-0.74,-0.34,1>,
<-0.74,-0.36,1>,
<-0.74,-0.38,1>,
<-0.74,-0.4,1>,
<-0.74,-0.42,1>,
<-0.74,-0.44,1>,
<-0.74,-0.46,1>,
<-0.74,-0.48,1>,
<-0.74,-0.5,1>,
<-0.74,-0.52,1>,
<-0.74,-0.54,1>,
<-0.74,-0.56,1>,
<-0.74,-0.58,1>,
<-0.74,-0.6,1>,
<-0.74,-0.62,1>,
<-0.74,-0.64,1>,
<-0.74,-0.66,1>,
<-0.74,-0.68,1>,
<-0.74,-0.7,1>,
<-0.74,-0.72,1>,
<-0.74,-0.74,1>,
<-0.74,-0.76,1>,
<-0.74,-0.78,1>,
<-0.74,-0.8,1>,
<-0.74,-0.82,1>,
<-0.74,-0.84,1>,
<-0.74,-0.86,1>,
<-0.74,-0.88,1>,
<-0.74,-0.9,1>,
<-0.74,-0.92,1>,
<-0.74,-0.94,1>,
<-0.74,-0.96,1>,
<-0.74,-0.98,1>,

<-0.74,-1,1>,
<-0.76,0,1>,
<-0.76,-0.02,1>,
<-0.76,-0.04,1>,
<-0.76,-0.06,1>,
<-0.76,-0.08,1>,
<-0.76,-0.1,1>,
<-0.76,-0.12,1>,
<-0.76,-0.14,1>,
<-0.76,-0.16,1>,
<-0.76,-0.18,1>,
<-0.76,-0.2,1>,
<-0.76,-0.22,1>,
<-0.76,-0.24,1>,
<-0.76,-0.26,1>,
<-0.76,-0.28,1>,
<-0.76,-0.3,1>,
<-0.76,-0.32,1>,
<-0.76,-0.34,1>,
<-0.76,-0.36,1>,
<-0.76,-0.38,1>,
<-0.76,-0.4,1>,
<-0.76,-0.42,1>,
<-0.76,-0.44,1>,
<-0.76,-0.46,1>,
<-0.76,-0.48,1>,
<-0.76,-0.5,1>,
<-0.76,-0.52,1>,
<-0.76,-0.54,1>,
<-0.76,-0.56,1>,
<-0.76,-0.58,1>,
<-0.76,-0.6,1>,
<-0.76,-0.62,1>,
<-0.76,-0.64,1>,
<-0.76,-0.66,1>,
<-0.76,-0.68,1>,
<-0.76,-0.7,1>,
<-0.76,-0.72,1>,
<-0.76,-0.74,1>,
<-0.76,-0.76,1>,
<-0.76,-0.78,1>,
<-0.76,-0.8,1>,
<-0.76,-0.82,1>,
<-0.76,-0.84,1>,
<-0.76,-0.86,1>,
<-0.76,-0.88,1>,
<-0.76,-0.9,1>,
<-0.76,-0.92,1>,
<-0.76,-0.94,1>,
<-0.76,-0.96,1>,
<-0.76,-0.98,1>,
<-0.76,-1,1>,
<-0.78,0,1>,
<-0.78,-0.02,1>,
<-0.78,-0.04,1>,
<-0.78,-0.06,1>,

<-0.78,-0.08,1>,
<-0.78,-0.1,1>,
<-0.78,-0.12,1>,
<-0.78,-0.14,1>,
<-0.78,-0.16,1>,
<-0.78,-0.18,1>,
<-0.78,-0.2,1>,
<-0.78,-0.22,1>,
<-0.78,-0.24,1>,
<-0.78,-0.26,1>,
<-0.78,-0.28,1>,
<-0.78,-0.3,1>,
<-0.78,-0.32,1>,
<-0.78,-0.34,1>,
<-0.78,-0.36,1>,
<-0.78,-0.38,1>,
<-0.78,-0.4,1>,
<-0.78,-0.42,1>,
<-0.78,-0.44,1>,
<-0.78,-0.46,1>,
<-0.78,-0.48,1>,
<-0.78,-0.5,1>,
<-0.78,-0.52,1>,
<-0.78,-0.54,1>,
<-0.78,-0.56,1>,
<-0.78,-0.58,1>,
<-0.78,-0.6,1>,
<-0.78,-0.62,1>,
<-0.78,-0.64,1>,
<-0.78,-0.66,1>,
<-0.78,-0.68,1>,
<-0.78,-0.7,1>,
<-0.78,-0.72,1>,
<-0.78,-0.74,1>,
<-0.78,-0.76,1>,
<-0.78,-0.78,1>,
<-0.78,-0.8,1>,
<-0.78,-0.82,1>,
<-0.78,-0.84,1>,
<-0.78,-0.86,1>,
<-0.78,-0.88,1>,
<-0.78,-0.9,1>,
<-0.78,-0.92,1>,
<-0.78,-0.94,1>,
<-0.78,-0.96,1>,
<-0.78,-0.98,1>,
<-0.78,-1,1>,
<-0.8,0,1>,
<-0.8,-0.02,1>,
<-0.8,-0.04,1>,
<-0.8,-0.06,1>,
<-0.8,-0.08,1>,
<-0.8,-0.1,1>,
<-0.8,-0.12,1>,
<-0.8,-0.14,1>,
<-0.8,-0.16,1>,

<-0.8,-0.18,1>,
<-0.8,-0.2,1>,
<-0.8,-0.22,1>,
<-0.8,-0.24,1>,
<-0.8,-0.26,1>,
<-0.8,-0.28,1>,
<-0.8,-0.3,1>,
<-0.8,-0.32,1>,
<-0.8,-0.34,1>,
<-0.8,-0.36,1>,
<-0.8,-0.38,1>,
<-0.8,-0.4,1>,
<-0.8,-0.42,1>,
<-0.8,-0.44,1>,
<-0.8,-0.46,1>,
<-0.8,-0.48,1>,
<-0.8,-0.5,1>,
<-0.8,-0.52,1>,
<-0.8,-0.54,1>,
<-0.8,-0.56,1>,
<-0.8,-0.58,1>,
<-0.8,-0.6,1>,
<-0.8,-0.62,1>,
<-0.8,-0.64,1>,
<-0.8,-0.66,1>,
<-0.8,-0.68,1>,
<-0.8,-0.7,1>,
<-0.8,-0.72,1>,
<-0.8,-0.74,1>,
<-0.8,-0.76,1>,
<-0.8,-0.78,1>,
<-0.8,-0.8,1>,
<-0.8,-0.82,1>,
<-0.8,-0.84,1>,
<-0.8,-0.86,1>,
<-0.8,-0.88,1>,
<-0.8,-0.9,1>,
<-0.8,-0.92,1>,
<-0.8,-0.94,1>,
<-0.8,-0.96,1>,
<-0.8,-0.98,1>,
<-0.8,-1,1>,
<-0.82,0,1>,
<-0.82,-0.02,1>,
<-0.82,-0.04,1>,
<-0.82,-0.06,1>,
<-0.82,-0.08,1>,
<-0.82,-0.1,1>,
<-0.82,-0.12,1>,
<-0.82,-0.14,1>,
<-0.82,-0.16,1>,
<-0.82,-0.18,1>,
<-0.82,-0.2,1>,
<-0.82,-0.22,1>,
<-0.82,-0.24,1>,
<-0.82,-0.26,1>,

<-0.82,-0.28,1>,
<-0.82,-0.3,1>,
<-0.82,-0.32,1>,
<-0.82,-0.34,1>,
<-0.82,-0.36,1>,
<-0.82,-0.38,1>,
<-0.82,-0.4,1>,
<-0.82,-0.42,1>,
<-0.82,-0.44,1>,
<-0.82,-0.46,1>,
<-0.82,-0.48,1>,
<-0.82,-0.5,1>,
<-0.82,-0.52,1>,
<-0.82,-0.54,1>,
<-0.82,-0.56,1>,
<-0.82,-0.58,1>,
<-0.82,-0.6,1>,
<-0.82,-0.62,1>,
<-0.82,-0.64,1>,
<-0.82,-0.66,1>,
<-0.82,-0.68,1>,
<-0.82,-0.7,1>,
<-0.82,-0.72,1>,
<-0.82,-0.74,1>,
<-0.82,-0.76,1>,
<-0.82,-0.78,1>,
<-0.82,-0.8,1>,
<-0.82,-0.82,1>,
<-0.82,-0.84,1>,
<-0.82,-0.86,1>,
<-0.82,-0.88,1>,
<-0.82,-0.9,1>,
<-0.82,-0.92,1>,
<-0.82,-0.94,1>,
<-0.82,-0.96,1>,
<-0.82,-0.98,1>,
<-0.82,-1,1>,
<-0.84,0,1>,
<-0.84,-0.02,1>,
<-0.84,-0.04,1>,
<-0.84,-0.06,1>,
<-0.84,-0.08,1>,
<-0.84,-0.1,1>,
<-0.84,-0.12,1>,
<-0.84,-0.14,1>,
<-0.84,-0.16,1>,
<-0.84,-0.18,1>,
<-0.84,-0.2,1>,
<-0.84,-0.22,1>,
<-0.84,-0.24,1>,
<-0.84,-0.26,1>,
<-0.84,-0.28,1>,
<-0.84,-0.3,1>,
<-0.84,-0.32,1>,
<-0.84,-0.34,1>,
<-0.84,-0.36,1>,

<-0.84,-0.38,1>,
<-0.84,-0.4,1>,
<-0.84,-0.42,1>,
<-0.84,-0.44,1>,
<-0.84,-0.46,1>,
<-0.84,-0.48,1>,
<-0.84,-0.5,1>,
<-0.84,-0.52,1>,
<-0.84,-0.54,1>,
<-0.84,-0.56,1>,
<-0.84,-0.58,1>,
<-0.84,-0.6,1>,
<-0.84,-0.62,1>,
<-0.84,-0.64,1>,
<-0.84,-0.66,1>,
<-0.84,-0.68,1>,
<-0.84,-0.7,1>,
<-0.84,-0.72,1>,
<-0.84,-0.74,1>,
<-0.84,-0.76,1>,
<-0.84,-0.78,1>,
<-0.84,-0.8,1>,
<-0.84,-0.82,1>,
<-0.84,-0.84,1>,
<-0.84,-0.86,1>,
<-0.84,-0.88,1>,
<-0.84,-0.9,1>,
<-0.84,-0.92,1>,
<-0.84,-0.94,1>,
<-0.84,-0.96,1>,
<-0.84,-0.98,1>,
<-0.84,-1,1>,
<-0.86,0,1>,
<-0.86,-0.02,1>,
<-0.86,-0.04,1>,
<-0.86,-0.06,1>,
<-0.86,-0.08,1>,
<-0.86,-0.1,1>,
<-0.86,-0.12,1>,
<-0.86,-0.14,1>,
<-0.86,-0.16,1>,
<-0.86,-0.18,1>,
<-0.86,-0.2,1>,
<-0.86,-0.22,1>,
<-0.86,-0.24,1>,
<-0.86,-0.26,1>,
<-0.86,-0.28,1>,
<-0.86,-0.3,1>,
<-0.86,-0.32,1>,
<-0.86,-0.34,1>,
<-0.86,-0.36,1>,
<-0.86,-0.38,1>,
<-0.86,-0.4,1>,
<-0.86,-0.42,1>,
<-0.86,-0.44,1>,
<-0.86,-0.46,1>,

<-0.86,-0.48,1>,
<-0.86,-0.5,1>,
<-0.86,-0.52,1>,
<-0.86,-0.54,1>,
<-0.86,-0.56,1>,
<-0.86,-0.58,1>,
<-0.86,-0.6,1>,
<-0.86,-0.62,1>,
<-0.86,-0.64,1>,
<-0.86,-0.66,1>,
<-0.86,-0.68,1>,
<-0.86,-0.7,1>,
<-0.86,-0.72,1>,
<-0.86,-0.74,1>,
<-0.86,-0.76,1>,
<-0.86,-0.78,1>,
<-0.86,-0.8,1>,
<-0.86,-0.82,1>,
<-0.86,-0.84,1>,
<-0.86,-0.86,1>,
<-0.86,-0.88,1>,
<-0.86,-0.9,1>,
<-0.86,-0.92,1>,
<-0.86,-0.94,1>,
<-0.86,-0.96,1>,
<-0.86,-0.98,1>,
<-0.86,-1,1>,
<-0.88,0,1>,
<-0.88,-0.02,1>,
<-0.88,-0.04,1>,
<-0.88,-0.06,1>,
<-0.88,-0.08,1>,
<-0.88,-0.1,1>,
<-0.88,-0.12,1>,
<-0.88,-0.14,1>,
<-0.88,-0.16,1>,
<-0.88,-0.18,1>,
<-0.88,-0.2,1>,
<-0.88,-0.22,1>,
<-0.88,-0.24,1>,
<-0.88,-0.26,1>,
<-0.88,-0.28,1>,
<-0.88,-0.3,1>,
<-0.88,-0.32,1>,
<-0.88,-0.34,1>,
<-0.88,-0.36,1>,
<-0.88,-0.38,1>,
<-0.88,-0.4,1>,
<-0.88,-0.42,1>,
<-0.88,-0.44,1>,
<-0.88,-0.46,1>,
<-0.88,-0.48,1>,
<-0.88,-0.5,1>,
<-0.88,-0.52,1>,
<-0.88,-0.54,1>,
<-0.88,-0.56,1>,

<-0.88,-0.58,1>,
<-0.88,-0.6,1>,
<-0.88,-0.62,1>,
<-0.88,-0.64,1>,
<-0.88,-0.66,1>,
<-0.88,-0.68,1>,
<-0.88,-0.7,1>,
<-0.88,-0.72,1>,
<-0.88,-0.74,1>,
<-0.88,-0.76,1>,
<-0.88,-0.78,1>,
<-0.88,-0.8,1>,
<-0.88,-0.82,1>,
<-0.88,-0.84,1>,
<-0.88,-0.86,1>,
<-0.88,-0.88,1>,
<-0.88,-0.9,1>,
<-0.88,-0.92,1>,
<-0.88,-0.94,1>,
<-0.88,-0.96,1>,
<-0.88,-0.98,1>,
<-0.88,-1,1>,
<-0.9,0,1>,
<-0.9,-0.02,1>,
<-0.9,-0.04,1>,
<-0.9,-0.06,1>,
<-0.9,-0.08,1>,
<-0.9,-0.1,1>,
<-0.9,-0.12,1>,
<-0.9,-0.14,1>,
<-0.9,-0.16,1>,
<-0.9,-0.18,1>,
<-0.9,-0.2,1>,
<-0.9,-0.22,1>,
<-0.9,-0.24,1>,
<-0.9,-0.26,1>,
<-0.9,-0.28,1>,
<-0.9,-0.3,1>,
<-0.9,-0.32,1>,
<-0.9,-0.34,1>,
<-0.9,-0.36,1>,
<-0.9,-0.38,1>,
<-0.9,-0.4,1>,
<-0.9,-0.42,1>,
<-0.9,-0.44,1>,
<-0.9,-0.46,1>,
<-0.9,-0.48,1>,
<-0.9,-0.5,1>,
<-0.9,-0.52,1>,
<-0.9,-0.54,1>,
<-0.9,-0.56,1>,
<-0.9,-0.58,1>,
<-0.9,-0.6,1>,
<-0.9,-0.62,1>,
<-0.9,-0.64,1>,
<-0.9,-0.66,1>,

<-0.9,-0.68,1>,
<-0.9,-0.7,1>,
<-0.9,-0.72,1>,
<-0.9,-0.74,1>,
<-0.9,-0.76,1>,
<-0.9,-0.78,1>,
<-0.9,-0.8,1>,
<-0.9,-0.82,1>,
<-0.9,-0.84,1>,
<-0.9,-0.86,1>,
<-0.9,-0.88,1>,
<-0.9,-0.9,1>,
<-0.9,-0.92,1>,
<-0.9,-0.94,1>,
<-0.9,-0.96,1>,
<-0.9,-0.98,1>,
<-0.9,-1,1>,
<-0.92,0,1>,
<-0.92,-0.02,1>,
<-0.92,-0.04,1>,
<-0.92,-0.06,1>,
<-0.92,-0.08,1>,
<-0.92,-0.1,1>,
<-0.92,-0.12,1>,
<-0.92,-0.14,1>,
<-0.92,-0.16,1>,
<-0.92,-0.18,1>,
<-0.92,-0.2,1>,
<-0.92,-0.22,1>,
<-0.92,-0.24,1>,
<-0.92,-0.26,1>,
<-0.92,-0.28,1>,
<-0.92,-0.3,1>,
<-0.92,-0.32,1>,
<-0.92,-0.34,1>,
<-0.92,-0.36,1>,
<-0.92,-0.38,1>,
<-0.92,-0.4,1>,
<-0.92,-0.42,1>,
<-0.92,-0.44,1>,
<-0.92,-0.46,1>,
<-0.92,-0.48,1>,
<-0.92,-0.5,1>,
<-0.92,-0.52,1>,
<-0.92,-0.54,1>,
<-0.92,-0.56,1>,
<-0.92,-0.58,1>,
<-0.92,-0.6,1>,
<-0.92,-0.62,1>,
<-0.92,-0.64,1>,
<-0.92,-0.66,1>,
<-0.92,-0.68,1>,
<-0.92,-0.7,1>,
<-0.92,-0.72,1>,
<-0.92,-0.74,1>,
<-0.92,-0.76,1>,

<-0.92,-0.78,1>,
<-0.92,-0.8,1>,
<-0.92,-0.82,1>,
<-0.92,-0.84,1>,
<-0.92,-0.86,1>,
<-0.92,-0.88,1>,
<-0.92,-0.9,1>,
<-0.92,-0.92,1>,
<-0.92,-0.94,1>,
<-0.92,-0.96,1>,
<-0.92,-0.98,1>,
<-0.92,-1,1>,
<-0.94,0,1>,
<-0.94,-0.02,1>,
<-0.94,-0.04,1>,
<-0.94,-0.06,1>,
<-0.94,-0.08,1>,
<-0.94,-0.1,1>,
<-0.94,-0.12,1>,
<-0.94,-0.14,1>,
<-0.94,-0.16,1>,
<-0.94,-0.18,1>,
<-0.94,-0.2,1>,
<-0.94,-0.22,1>,
<-0.94,-0.24,1>,
<-0.94,-0.26,1>,
<-0.94,-0.28,1>,
<-0.94,-0.3,1>,
<-0.94,-0.32,1>,
<-0.94,-0.34,1>,
<-0.94,-0.36,1>,
<-0.94,-0.38,1>,
<-0.94,-0.4,1>,
<-0.94,-0.42,1>,
<-0.94,-0.44,1>,
<-0.94,-0.46,1>,
<-0.94,-0.48,1>,
<-0.94,-0.5,1>,
<-0.94,-0.52,1>,
<-0.94,-0.54,1>,
<-0.94,-0.56,1>,
<-0.94,-0.58,1>,
<-0.94,-0.6,1>,
<-0.94,-0.62,1>,
<-0.94,-0.64,1>,
<-0.94,-0.66,1>,
<-0.94,-0.68,1>,
<-0.94,-0.7,1>,
<-0.94,-0.72,1>,
<-0.94,-0.74,1>,
<-0.94,-0.76,1>,
<-0.94,-0.78,1>,
<-0.94,-0.8,1>,
<-0.94,-0.82,1>,
<-0.94,-0.84,1>,
<-0.94,-0.86,1>,

<-0.94,-0.88,1>,
<-0.94,-0.9,1>,
<-0.94,-0.92,1>,
<-0.94,-0.94,1>,
<-0.94,-0.96,1>,
<-0.94,-0.98,1>,
<-0.94,-1,1>,
<-0.96,0,1>,
<-0.96,-0.02,1>,
<-0.96,-0.04,1>,
<-0.96,-0.06,1>,
<-0.96,-0.08,1>,
<-0.96,-0.1,1>,
<-0.96,-0.12,1>,
<-0.96,-0.14,1>,
<-0.96,-0.16,1>,
<-0.96,-0.18,1>,
<-0.96,-0.2,1>,
<-0.96,-0.22,1>,
<-0.96,-0.24,1>,
<-0.96,-0.26,1>,
<-0.96,-0.28,1>,
<-0.96,-0.3,1>,
<-0.96,-0.32,1>,
<-0.96,-0.34,1>,
<-0.96,-0.36,1>,
<-0.96,-0.38,1>,
<-0.96,-0.4,1>,
<-0.96,-0.42,1>,
<-0.96,-0.44,1>,
<-0.96,-0.46,1>,
<-0.96,-0.48,1>,
<-0.96,-0.5,1>,
<-0.96,-0.52,1>,
<-0.96,-0.54,1>,
<-0.96,-0.56,1>,
<-0.96,-0.58,1>,
<-0.96,-0.6,1>,
<-0.96,-0.62,1>,
<-0.96,-0.64,1>,
<-0.96,-0.66,1>,
<-0.96,-0.68,1>,
<-0.96,-0.7,1>,
<-0.96,-0.72,1>,
<-0.96,-0.74,1>,
<-0.96,-0.76,1>,
<-0.96,-0.78,1>,
<-0.96,-0.8,1>,
<-0.96,-0.82,1>,
<-0.96,-0.84,1>,
<-0.96,-0.86,1>,
<-0.96,-0.88,1>,
<-0.96,-0.9,1>,
<-0.96,-0.92,1>,
<-0.96,-0.94,1>,
<-0.96,-0.96,1>,

<-0.96,-0.98,1>,
<-0.96,-1,1>,
<-0.98,0,1>,
<-0.98,-0.02,1>,
<-0.98,-0.04,1>,
<-0.98,-0.06,1>,
<-0.98,-0.08,1>,
<-0.98,-0.1,1>,
<-0.98,-0.12,1>,
<-0.98,-0.14,1>,
<-0.98,-0.16,1>,
<-0.98,-0.18,1>,
<-0.98,-0.2,1>,
<-0.98,-0.22,1>,
<-0.98,-0.24,1>,
<-0.98,-0.26,1>,
<-0.98,-0.28,1>,
<-0.98,-0.3,1>,
<-0.98,-0.32,1>,
<-0.98,-0.34,1>,
<-0.98,-0.36,1>,
<-0.98,-0.38,1>,
<-0.98,-0.4,1>,
<-0.98,-0.42,1>,
<-0.98,-0.44,1>,
<-0.98,-0.46,1>,
<-0.98,-0.48,1>,
<-0.98,-0.5,1>,
<-0.98,-0.52,1>,
<-0.98,-0.54,1>,
<-0.98,-0.56,1>,
<-0.98,-0.58,1>,
<-0.98,-0.6,1>,
<-0.98,-0.62,1>,
<-0.98,-0.64,1>,
<-0.98,-0.66,1>,
<-0.98,-0.68,1>,
<-0.98,-0.7,1>,
<-0.98,-0.72,1>,
<-0.98,-0.74,1>,
<-0.98,-0.76,1>,
<-0.98,-0.78,1>,
<-0.98,-0.8,1>,
<-0.98,-0.82,1>,
<-0.98,-0.84,1>,
<-0.98,-0.86,1>,
<-0.98,-0.88,1>,
<-0.98,-0.9,1>,
<-0.98,-0.92,1>,
<-0.98,-0.94,1>,
<-0.98,-0.96,1>,
<-0.98,-0.98,1>,
<-0.98,-1,1>,
<-1,0,1>,
<-1,-0.02,1>,
<-1,-0.04,1>,

```

<-1,-0.06,1>,
<-1,-0.08,1>,
<-1,-0.1,1>,
<-1,-0.12,1>,
<-1,-0.14,1>,
<-1,-0.16,1>,
<-1,-0.18,1>,
<-1,-0.2,1>,
<-1,-0.22,1>,
<-1,-0.24,1>,
<-1,-0.26,1>,
<-1,-0.28,1>,
<-1,-0.3,1>,
<-1,-0.32,1>,
<-1,-0.34,1>,
<-1,-0.36,1>,
<-1,-0.38,1>,
<-1,-0.4,1>,
<-1,-0.42,1>,
<-1,-0.44,1>,
<-1,-0.46,1>,
<-1,-0.48,1>,
<-1,-0.5,1>,
<-1,-0.52,1>,
<-1,-0.54,1>,
<-1,-0.56,1>,
<-1,-0.58,1>,
<-1,-0.6,1>,
<-1,-0.62,1>,
<-1,-0.64,1>,
<-1,-0.66,1>,
<-1,-0.68,1>,
<-1,-0.7,1>,
<-1,-0.72,1>,
<-1,-0.74,1>,
<-1,-0.76,1>,
<-1,-0.78,1>,
<-1,-0.8,1>,
<-1,-0.82,1>,
<-1,-0.84,1>,
<-1,-0.86,1>,
<-1,-0.88,1>,
<-1,-0.9,1>,
<-1,-0.92,1>,
<-1,-0.94,1>,
<-1,-0.96,1>,
<-1,-0.98,1>,
<-1,-1,1>,
}

face_indices { 5000,
<0,1,51>,
<52,51,1>,
<1,2,52>,
<53,52,2>,
<2,3,53>,
<54,53,3>,

```

<3, 4, 54>,
<55, 54, 4>,
<4, 5, 55>,
<56, 55, 5>,
<5, 6, 56>,
<57, 56, 6>,
<6, 7, 57>,
<58, 57, 7>,
<7, 8, 58>,
<59, 58, 8>,
<8, 9, 59>,
<60, 59, 9>,
<9, 10, 60>,
<61, 60, 10>,
<10, 11, 61>,
<62, 61, 11>,
<11, 12, 62>,
<63, 62, 12>,
<12, 13, 63>,
<64, 63, 13>,
<13, 14, 64>,
<65, 64, 14>,
<14, 15, 65>,
<66, 65, 15>,
<15, 16, 66>,
<67, 66, 16>,
<16, 17, 67>,
<68, 67, 17>,
<17, 18, 68>,
<69, 68, 18>,
<18, 19, 69>,
<70, 69, 19>,
<19, 20, 70>,
<71, 70, 20>,
<20, 21, 71>,
<72, 71, 21>,
<21, 22, 72>,
<73, 72, 22>,
<22, 23, 73>,
<74, 73, 23>,
<23, 24, 74>,
<75, 74, 24>,
<24, 25, 75>,
<76, 75, 25>,
<25, 26, 76>,
<77, 76, 26>,
<26, 27, 77>,
<78, 77, 27>,
<27, 28, 78>,
<79, 78, 28>,
<28, 29, 79>,
<80, 79, 29>,
<29, 30, 80>,
<81, 80, 30>,
<30, 31, 81>,
<82, 81, 31>,

<31, 32, 82>,
<83, 82, 32>,
<32, 33, 83>,
<84, 83, 33>,
<33, 34, 84>,
<85, 84, 34>,
<34, 35, 85>,
<86, 85, 35>,
<35, 36, 86>,
<87, 86, 36>,
<36, 37, 87>,
<88, 87, 37>,
<37, 38, 88>,
<89, 88, 38>,
<38, 39, 89>,
<90, 89, 39>,
<39, 40, 90>,
<91, 90, 40>,
<40, 41, 91>,
<92, 91, 41>,
<41, 42, 92>,
<93, 92, 42>,
<42, 43, 93>,
<94, 93, 43>,
<43, 44, 94>,
<95, 94, 44>,
<44, 45, 95>,
<96, 95, 45>,
<45, 46, 96>,
<97, 96, 46>,
<46, 47, 97>,
<98, 97, 47>,
<47, 48, 98>,
<99, 98, 48>,
<48, 49, 99>,
<100, 99, 49>,
<49, 50, 100>,
<101, 100, 50>,
<51, 52, 102>,
<103, 102, 52>,
<52, 53, 103>,
<104, 103, 53>,
<53, 54, 104>,
<105, 104, 54>,
<54, 55, 105>,
<106, 105, 55>,
<55, 56, 106>,
<107, 106, 56>,
<56, 57, 107>,
<108, 107, 57>,
<57, 58, 108>,
<109, 108, 58>,
<58, 59, 109>,
<110, 109, 59>,
<59, 60, 110>,
<111, 110, 60>,

<60, 61, 111>,
<112, 111, 61>,
<61, 62, 112>,
<113, 112, 62>,
<62, 63, 113>,
<114, 113, 63>,
<63, 64, 114>,
<115, 114, 64>,
<64, 65, 115>,
<116, 115, 65>,
<65, 66, 116>,
<117, 116, 66>,
<66, 67, 117>,
<118, 117, 67>,
<67, 68, 118>,
<119, 118, 68>,
<68, 69, 119>,
<120, 119, 69>,
<69, 70, 120>,
<121, 120, 70>,
<70, 71, 121>,
<122, 121, 71>,
<71, 72, 122>,
<123, 122, 72>,
<72, 73, 123>,
<124, 123, 73>,
<73, 74, 124>,
<125, 124, 74>,
<74, 75, 125>,
<126, 125, 75>,
<75, 76, 126>,
<127, 126, 76>,
<76, 77, 127>,
<128, 127, 77>,
<77, 78, 128>,
<129, 128, 78>,
<78, 79, 129>,
<130, 129, 79>,
<79, 80, 130>,
<131, 130, 80>,
<80, 81, 131>,
<132, 131, 81>,
<81, 82, 132>,
<133, 132, 82>,
<82, 83, 133>,
<134, 133, 83>,
<83, 84, 134>,
<135, 134, 84>,
<84, 85, 135>,
<136, 135, 85>,
<85, 86, 136>,
<137, 136, 86>,
<86, 87, 137>,
<138, 137, 87>,
<87, 88, 138>,
<139, 138, 88>,

<88,89,139>,
<140,139,89>,
<89,90,140>,
<141,140,90>,
<90,91,141>,
<142,141,91>,
<91,92,142>,
<143,142,92>,
<92,93,143>,
<144,143,93>,
<93,94,144>,
<145,144,94>,
<94,95,145>,
<146,145,95>,
<95,96,146>,
<147,146,96>,
<96,97,147>,
<148,147,97>,
<97,98,148>,
<149,148,98>,
<98,99,149>,
<150,149,99>,
<99,100,150>,
<151,150,100>,
<100,101,151>,
<152,151,101>,
<102,103,153>,
<154,153,103>,
<103,104,154>,
<155,154,104>,
<104,105,155>,
<156,155,105>,
<105,106,156>,
<157,156,106>,
<106,107,157>,
<158,157,107>,
<107,108,158>,
<159,158,108>,
<108,109,159>,
<160,159,109>,
<109,110,160>,
<161,160,110>,
<110,111,161>,
<162,161,111>,
<111,112,162>,
<163,162,112>,
<112,113,163>,
<164,163,113>,
<113,114,164>,
<165,164,114>,
<114,115,165>,
<166,165,115>,
<115,116,166>,
<167,166,116>,
<116,117,167>,
<168,167,117>,

<117,118,168>,
<169,168,118>,
<118,119,169>,
<170,169,119>,
<119,120,170>,
<171,170,120>,
<120,121,171>,
<172,171,121>,
<121,122,172>,
<173,172,122>,
<122,123,173>,
<174,173,123>,
<123,124,174>,
<175,174,124>,
<124,125,175>,
<176,175,125>,
<125,126,176>,
<177,176,126>,
<126,127,177>,
<178,177,127>,
<127,128,178>,
<179,178,128>,
<128,129,179>,
<180,179,129>,
<129,130,180>,
<181,180,130>,
<130,131,181>,
<182,181,131>,
<131,132,182>,
<183,182,132>,
<132,133,183>,
<184,183,133>,
<133,134,184>,
<185,184,134>,
<134,135,185>,
<186,185,135>,
<135,136,186>,
<187,186,136>,
<136,137,187>,
<188,187,137>,
<137,138,188>,
<189,188,138>,
<138,139,189>,
<190,189,139>,
<139,140,190>,
<191,190,140>,
<140,141,191>,
<192,191,141>,
<141,142,192>,
<193,192,142>,
<142,143,193>,
<194,193,143>,
<143,144,194>,
<195,194,144>,
<144,145,195>,
<196,195,145>,

<145,146,196>,
<197,196,146>,
<146,147,197>,
<198,197,147>,
<147,148,198>,
<199,198,148>,
<148,149,199>,
<200,199,149>,
<149,150,200>,
<201,200,150>,
<150,151,201>,
<202,201,151>,
<151,152,202>,
<203,202,152>,
<153,154,204>,
<205,204,154>,
<154,155,205>,
<206,205,155>,
<155,156,206>,
<207,206,156>,
<156,157,207>,
<208,207,157>,
<157,158,208>,
<209,208,158>,
<158,159,209>,
<210,209,159>,
<159,160,210>,
<211,210,160>,
<160,161,211>,
<212,211,161>,
<161,162,212>,
<213,212,162>,
<162,163,213>,
<214,213,163>,
<163,164,214>,
<215,214,164>,
<164,165,215>,
<216,215,165>,
<165,166,216>,
<217,216,166>,
<166,167,217>,
<218,217,167>,
<167,168,218>,
<219,218,168>,
<168,169,219>,
<220,219,169>,
<169,170,220>,
<221,220,170>,
<170,171,221>,
<222,221,171>,
<171,172,222>,
<223,222,172>,
<172,173,223>,
<224,223,173>,
<173,174,224>,
<225,224,174>,

<174,175,225>,
<226,225,175>,
<175,176,226>,
<227,226,176>,
<176,177,227>,
<228,227,177>,
<177,178,228>,
<229,228,178>,
<178,179,229>,
<230,229,179>,
<179,180,230>,
<231,230,180>,
<180,181,231>,
<232,231,181>,
<181,182,232>,
<233,232,182>,
<182,183,233>,
<234,233,183>,
<183,184,234>,
<235,234,184>,
<184,185,235>,
<236,235,185>,
<185,186,236>,
<237,236,186>,
<186,187,237>,
<238,237,187>,
<187,188,238>,
<239,238,188>,
<188,189,239>,
<240,239,189>,
<189,190,240>,
<241,240,190>,
<190,191,241>,
<242,241,191>,
<191,192,242>,
<243,242,192>,
<192,193,243>,
<244,243,193>,
<193,194,244>,
<245,244,194>,
<194,195,245>,
<246,245,195>,
<195,196,246>,
<247,246,196>,
<196,197,247>,
<248,247,197>,
<197,198,248>,
<249,248,198>,
<198,199,249>,
<250,249,199>,
<199,200,250>,
<251,250,200>,
<200,201,251>,
<252,251,201>,
<201,202,252>,
<253,252,202>,

<202,203,253>,
<254,253,203>,
<204,205,255>,
<256,255,205>,
<205,206,256>,
<257,256,206>,
<206,207,257>,
<258,257,207>,
<207,208,258>,
<259,258,208>,
<208,209,259>,
<260,259,209>,
<209,210,260>,
<261,260,210>,
<210,211,261>,
<262,261,211>,
<211,212,262>,
<263,262,212>,
<212,213,263>,
<264,263,213>,
<213,214,264>,
<265,264,214>,
<214,215,265>,
<266,265,215>,
<215,216,266>,
<267,266,216>,
<216,217,267>,
<268,267,217>,
<217,218,268>,
<269,268,218>,
<218,219,269>,
<270,269,219>,
<219,220,270>,
<271,270,220>,
<220,221,271>,
<272,271,221>,
<221,222,272>,
<273,272,222>,
<222,223,273>,
<274,273,223>,
<223,224,274>,
<275,274,224>,
<224,225,275>,
<276,275,225>,
<225,226,276>,
<277,276,226>,
<226,227,277>,
<278,277,227>,
<227,228,278>,
<279,278,228>,
<228,229,279>,
<280,279,229>,
<229,230,280>,
<281,280,230>,
<230,231,281>,
<282,281,231>,

<231,232,282>,
<283,282,232>,
<232,233,283>,
<284,283,233>,
<233,234,284>,
<285,284,234>,
<234,235,285>,
<286,285,235>,
<235,236,286>,
<287,286,236>,
<236,237,287>,
<288,287,237>,
<237,238,288>,
<289,288,238>,
<238,239,289>,
<290,289,239>,
<239,240,290>,
<291,290,240>,
<240,241,291>,
<292,291,241>,
<241,242,292>,
<293,292,242>,
<242,243,293>,
<294,293,243>,
<243,244,294>,
<295,294,244>,
<244,245,295>,
<296,295,245>,
<245,246,296>,
<297,296,246>,
<246,247,297>,
<298,297,247>,
<247,248,298>,
<299,298,248>,
<248,249,299>,
<300,299,249>,
<249,250,300>,
<301,300,250>,
<250,251,301>,
<302,301,251>,
<251,252,302>,
<303,302,252>,
<252,253,303>,
<304,303,253>,
<253,254,304>,
<305,304,254>,
<255,256,306>,
<307,306,256>,
<256,257,307>,
<308,307,257>,
<257,258,308>,
<309,308,258>,
<258,259,309>,
<310,309,259>,
<259,260,310>,
<311,310,260>,

<260, 261, 311>,
<312, 311, 261>,
<261, 262, 312>,
<313, 312, 262>,
<262, 263, 313>,
<314, 313, 263>,
<263, 264, 314>,
<315, 314, 264>,
<264, 265, 315>,
<316, 315, 265>,
<265, 266, 316>,
<317, 316, 266>,
<266, 267, 317>,
<318, 317, 267>,
<267, 268, 318>,
<319, 318, 268>,
<268, 269, 319>,
<320, 319, 269>,
<269, 270, 320>,
<321, 320, 270>,
<270, 271, 321>,
<322, 321, 271>,
<271, 272, 322>,
<323, 322, 272>,
<272, 273, 323>,
<324, 323, 273>,
<273, 274, 324>,
<325, 324, 274>,
<274, 275, 325>,
<326, 325, 275>,
<275, 276, 326>,
<327, 326, 276>,
<276, 277, 327>,
<328, 327, 277>,
<277, 278, 328>,
<329, 328, 278>,
<278, 279, 329>,
<330, 329, 279>,
<279, 280, 330>,
<331, 330, 280>,
<280, 281, 331>,
<332, 331, 281>,
<281, 282, 332>,
<333, 332, 282>,
<282, 283, 333>,
<334, 333, 283>,
<283, 284, 334>,
<335, 334, 284>,
<284, 285, 335>,
<336, 335, 285>,
<285, 286, 336>,
<337, 336, 286>,
<286, 287, 337>,
<338, 337, 287>,
<287, 288, 338>,
<339, 338, 288>,

<288, 289, 339>,
<340, 339, 289>,
<289, 290, 340>,
<341, 340, 290>,
<290, 291, 341>,
<342, 341, 291>,
<291, 292, 342>,
<343, 342, 292>,
<292, 293, 343>,
<344, 343, 293>,
<293, 294, 344>,
<345, 344, 294>,
<294, 295, 345>,
<346, 345, 295>,
<295, 296, 346>,
<347, 346, 296>,
<296, 297, 347>,
<348, 347, 297>,
<297, 298, 348>,
<349, 348, 298>,
<298, 299, 349>,
<350, 349, 299>,
<299, 300, 350>,
<351, 350, 300>,
<300, 301, 351>,
<352, 351, 301>,
<301, 302, 352>,
<353, 352, 302>,
<302, 303, 353>,
<354, 353, 303>,
<303, 304, 354>,
<355, 354, 304>,
<304, 305, 355>,
<356, 355, 305>,
<306, 307, 357>,
<358, 357, 307>,
<307, 308, 358>,
<359, 358, 308>,
<308, 309, 359>,
<360, 359, 309>,
<309, 310, 360>,
<361, 360, 310>,
<310, 311, 361>,
<362, 361, 311>,
<311, 312, 362>,
<363, 362, 312>,
<312, 313, 363>,
<364, 363, 313>,
<313, 314, 364>,
<365, 364, 314>,
<314, 315, 365>,
<366, 365, 315>,
<315, 316, 366>,
<367, 366, 316>,
<316, 317, 367>,
<368, 367, 317>,

<317, 318, 368>,
<369, 368, 318>,
<318, 319, 369>,
<370, 369, 319>,
<319, 320, 370>,
<371, 370, 320>,
<320, 321, 371>,
<372, 371, 321>,
<321, 322, 372>,
<373, 372, 322>,
<322, 323, 373>,
<374, 373, 323>,
<323, 324, 374>,
<375, 374, 324>,
<324, 325, 375>,
<376, 375, 325>,
<325, 326, 376>,
<377, 376, 326>,
<326, 327, 377>,
<378, 377, 327>,
<327, 328, 378>,
<379, 378, 328>,
<328, 329, 379>,
<380, 379, 329>,
<329, 330, 380>,
<381, 380, 330>,
<330, 331, 381>,
<382, 381, 331>,
<331, 332, 382>,
<383, 382, 332>,
<332, 333, 383>,
<384, 383, 333>,
<333, 334, 384>,
<385, 384, 334>,
<334, 335, 385>,
<386, 385, 335>,
<335, 336, 386>,
<387, 386, 336>,
<336, 337, 387>,
<388, 387, 337>,
<337, 338, 388>,
<389, 388, 338>,
<338, 339, 389>,
<390, 389, 339>,
<339, 340, 390>,
<391, 390, 340>,
<340, 341, 391>,
<392, 391, 341>,
<341, 342, 392>,
<393, 392, 342>,
<342, 343, 393>,
<394, 393, 343>,
<343, 344, 394>,
<395, 394, 344>,
<344, 345, 395>,
<396, 395, 345>,

<345, 346, 396>,
<397, 396, 346>,
<346, 347, 397>,
<398, 397, 347>,
<347, 348, 398>,
<399, 398, 348>,
<348, 349, 399>,
<400, 399, 349>,
<349, 350, 400>,
<401, 400, 350>,
<350, 351, 401>,
<402, 401, 351>,
<351, 352, 402>,
<403, 402, 352>,
<352, 353, 403>,
<404, 403, 353>,
<353, 354, 404>,
<405, 404, 354>,
<354, 355, 405>,
<406, 405, 355>,
<355, 356, 406>,
<407, 406, 356>,
<357, 358, 408>,
<409, 408, 358>,
<358, 359, 409>,
<410, 409, 359>,
<359, 360, 410>,
<411, 410, 360>,
<360, 361, 411>,
<412, 411, 361>,
<361, 362, 412>,
<413, 412, 362>,
<362, 363, 413>,
<414, 413, 363>,
<363, 364, 414>,
<415, 414, 364>,
<364, 365, 415>,
<416, 415, 365>,
<365, 366, 416>,
<417, 416, 366>,
<366, 367, 417>,
<418, 417, 367>,
<367, 368, 418>,
<419, 418, 368>,
<368, 369, 419>,
<420, 419, 369>,
<369, 370, 420>,
<421, 420, 370>,
<370, 371, 421>,
<422, 421, 371>,
<371, 372, 422>,
<423, 422, 372>,
<372, 373, 423>,
<424, 423, 373>,
<373, 374, 424>,
<425, 424, 374>,

<374, 375, 425>,
<426, 425, 375>,
<375, 376, 426>,
<427, 426, 376>,
<376, 377, 427>,
<428, 427, 377>,
<377, 378, 428>,
<429, 428, 378>,
<378, 379, 429>,
<430, 429, 379>,
<379, 380, 430>,
<431, 430, 380>,
<380, 381, 431>,
<432, 431, 381>,
<381, 382, 432>,
<433, 432, 382>,
<382, 383, 433>,
<434, 433, 383>,
<383, 384, 434>,
<435, 434, 384>,
<384, 385, 435>,
<436, 435, 385>,
<385, 386, 436>,
<437, 436, 386>,
<386, 387, 437>,
<438, 437, 387>,
<387, 388, 438>,
<439, 438, 388>,
<388, 389, 439>,
<440, 439, 389>,
<389, 390, 440>,
<441, 440, 390>,
<390, 391, 441>,
<442, 441, 391>,
<391, 392, 442>,
<443, 442, 392>,
<392, 393, 443>,
<444, 443, 393>,
<393, 394, 444>,
<445, 444, 394>,
<394, 395, 445>,
<446, 445, 395>,
<395, 396, 446>,
<447, 446, 396>,
<396, 397, 447>,
<448, 447, 397>,
<397, 398, 448>,
<449, 448, 398>,
<398, 399, 449>,
<450, 449, 399>,
<399, 400, 450>,
<451, 450, 400>,
<400, 401, 451>,
<452, 451, 401>,
<401, 402, 452>,
<453, 452, 402>,

<402, 403, 453>,
<454, 453, 403>,
<403, 404, 454>,
<455, 454, 404>,
<404, 405, 455>,
<456, 455, 405>,
<405, 406, 456>,
<457, 456, 406>,
<406, 407, 457>,
<458, 457, 407>,
<408, 409, 459>,
<460, 459, 409>,
<409, 410, 460>,
<461, 460, 410>,
<410, 411, 461>,
<462, 461, 411>,
<411, 412, 462>,
<463, 462, 412>,
<412, 413, 463>,
<464, 463, 413>,
<413, 414, 464>,
<465, 464, 414>,
<414, 415, 465>,
<466, 465, 415>,
<415, 416, 466>,
<467, 466, 416>,
<416, 417, 467>,
<468, 467, 417>,
<417, 418, 468>,
<469, 468, 418>,
<418, 419, 469>,
<470, 469, 419>,
<419, 420, 470>,
<471, 470, 420>,
<420, 421, 471>,
<472, 471, 421>,
<421, 422, 472>,
<473, 472, 422>,
<422, 423, 473>,
<474, 473, 423>,
<423, 424, 474>,
<475, 474, 424>,
<424, 425, 475>,
<476, 475, 425>,
<425, 426, 476>,
<477, 476, 426>,
<426, 427, 477>,
<478, 477, 427>,
<427, 428, 478>,
<479, 478, 428>,
<428, 429, 479>,
<480, 479, 429>,
<429, 430, 480>,
<481, 480, 430>,
<430, 431, 481>,
<482, 481, 431>,

<431, 432, 482>,
<483, 482, 432>,
<432, 433, 483>,
<484, 483, 433>,
<433, 434, 484>,
<485, 484, 434>,
<434, 435, 485>,
<486, 485, 435>,
<435, 436, 486>,
<487, 486, 436>,
<436, 437, 487>,
<488, 487, 437>,
<437, 438, 488>,
<489, 488, 438>,
<438, 439, 489>,
<490, 489, 439>,
<439, 440, 490>,
<491, 490, 440>,
<440, 441, 491>,
<492, 491, 441>,
<441, 442, 492>,
<493, 492, 442>,
<442, 443, 493>,
<494, 493, 443>,
<443, 444, 494>,
<495, 494, 444>,
<444, 445, 495>,
<496, 495, 445>,
<445, 446, 496>,
<497, 496, 446>,
<446, 447, 497>,
<498, 497, 447>,
<447, 448, 498>,
<499, 498, 448>,
<448, 449, 499>,
<500, 499, 449>,
<449, 450, 500>,
<501, 500, 450>,
<450, 451, 501>,
<502, 501, 451>,
<451, 452, 502>,
<503, 502, 452>,
<452, 453, 503>,
<504, 503, 453>,
<453, 454, 504>,
<505, 504, 454>,
<454, 455, 505>,
<506, 505, 455>,
<455, 456, 506>,
<507, 506, 456>,
<456, 457, 507>,
<508, 507, 457>,
<457, 458, 508>,
<509, 508, 458>,
<459, 460, 510>,
<511, 510, 460>,

<460, 461, 511>,
<512, 511, 461>,
<461, 462, 512>,
<513, 512, 462>,
<462, 463, 513>,
<514, 513, 463>,
<463, 464, 514>,
<515, 514, 464>,
<464, 465, 515>,
<516, 515, 465>,
<465, 466, 516>,
<517, 516, 466>,
<466, 467, 517>,
<518, 517, 467>,
<467, 468, 518>,
<519, 518, 468>,
<468, 469, 519>,
<520, 519, 469>,
<469, 470, 520>,
<521, 520, 470>,
<470, 471, 521>,
<522, 521, 471>,
<471, 472, 522>,
<523, 522, 472>,
<472, 473, 523>,
<524, 523, 473>,
<473, 474, 524>,
<525, 524, 474>,
<474, 475, 525>,
<526, 525, 475>,
<475, 476, 526>,
<527, 526, 476>,
<476, 477, 527>,
<528, 527, 477>,
<477, 478, 528>,
<529, 528, 478>,
<478, 479, 529>,
<530, 529, 479>,
<479, 480, 530>,
<531, 530, 480>,
<480, 481, 531>,
<532, 531, 481>,
<481, 482, 532>,
<533, 532, 482>,
<482, 483, 533>,
<534, 533, 483>,
<483, 484, 534>,
<535, 534, 484>,
<484, 485, 535>,
<536, 535, 485>,
<485, 486, 536>,
<537, 536, 486>,
<486, 487, 537>,
<538, 537, 487>,
<487, 488, 538>,
<539, 538, 488>,

<488, 489, 539>,
<540, 539, 489>,
<489, 490, 540>,
<541, 540, 490>,
<490, 491, 541>,
<542, 541, 491>,
<491, 492, 542>,
<543, 542, 492>,
<492, 493, 543>,
<544, 543, 493>,
<493, 494, 544>,
<545, 544, 494>,
<494, 495, 545>,
<546, 545, 495>,
<495, 496, 546>,
<547, 546, 496>,
<496, 497, 547>,
<548, 547, 497>,
<497, 498, 548>,
<549, 548, 498>,
<498, 499, 549>,
<550, 549, 499>,
<499, 500, 550>,
<551, 550, 500>,
<500, 501, 551>,
<552, 551, 501>,
<501, 502, 552>,
<553, 552, 502>,
<502, 503, 553>,
<554, 553, 503>,
<503, 504, 554>,
<555, 554, 504>,
<504, 505, 555>,
<556, 555, 505>,
<505, 506, 556>,
<557, 556, 506>,
<506, 507, 557>,
<558, 557, 507>,
<507, 508, 558>,
<559, 558, 508>,
<508, 509, 559>,
<560, 559, 509>,
<510, 511, 561>,
<562, 561, 511>,
<511, 512, 562>,
<563, 562, 512>,
<512, 513, 563>,
<564, 563, 513>,
<513, 514, 564>,
<565, 564, 514>,
<514, 515, 565>,
<566, 565, 515>,
<515, 516, 566>,
<567, 566, 516>,
<516, 517, 567>,
<568, 567, 517>,

<517, 518, 568>,
<569, 568, 518>,
<518, 519, 569>,
<570, 569, 519>,
<519, 520, 570>,
<571, 570, 520>,
<520, 521, 571>,
<572, 571, 521>,
<521, 522, 572>,
<573, 572, 522>,
<522, 523, 573>,
<574, 573, 523>,
<523, 524, 574>,
<575, 574, 524>,
<524, 525, 575>,
<576, 575, 525>,
<525, 526, 576>,
<577, 576, 526>,
<526, 527, 577>,
<578, 577, 527>,
<527, 528, 578>,
<579, 578, 528>,
<528, 529, 579>,
<580, 579, 529>,
<529, 530, 580>,
<581, 580, 530>,
<530, 531, 581>,
<582, 581, 531>,
<531, 532, 582>,
<583, 582, 532>,
<532, 533, 583>,
<584, 583, 533>,
<533, 534, 584>,
<585, 584, 534>,
<534, 535, 585>,
<586, 585, 535>,
<535, 536, 586>,
<587, 586, 536>,
<536, 537, 587>,
<588, 587, 537>,
<537, 538, 588>,
<589, 588, 538>,
<538, 539, 589>,
<590, 589, 539>,
<539, 540, 590>,
<591, 590, 540>,
<540, 541, 591>,
<592, 591, 541>,
<541, 542, 592>,
<593, 592, 542>,
<542, 543, 593>,
<594, 593, 543>,
<543, 544, 594>,
<595, 594, 544>,
<544, 545, 595>,
<596, 595, 545>,

<545, 546, 596>,
<597, 596, 546>,
<546, 547, 597>,
<598, 597, 547>,
<547, 548, 598>,
<599, 598, 548>,
<548, 549, 599>,
<600, 599, 549>,
<549, 550, 600>,
<601, 600, 550>,
<550, 551, 601>,
<602, 601, 551>,
<551, 552, 602>,
<603, 602, 552>,
<552, 553, 603>,
<604, 603, 553>,
<553, 554, 604>,
<605, 604, 554>,
<554, 555, 605>,
<606, 605, 555>,
<555, 556, 606>,
<607, 606, 556>,
<556, 557, 607>,
<608, 607, 557>,
<557, 558, 608>,
<609, 608, 558>,
<558, 559, 609>,
<610, 609, 559>,
<559, 560, 610>,
<611, 610, 560>,
<561, 562, 612>,
<613, 612, 562>,
<562, 563, 613>,
<614, 613, 563>,
<563, 564, 614>,
<615, 614, 564>,
<564, 565, 615>,
<616, 615, 565>,
<565, 566, 616>,
<617, 616, 566>,
<566, 567, 617>,
<618, 617, 567>,
<567, 568, 618>,
<619, 618, 568>,
<568, 569, 619>,
<620, 619, 569>,
<569, 570, 620>,
<621, 620, 570>,
<570, 571, 621>,
<622, 621, 571>,
<571, 572, 622>,
<623, 622, 572>,
<572, 573, 623>,
<624, 623, 573>,
<573, 574, 624>,
<625, 624, 574>,

<574, 575, 625>,
<626, 625, 575>,
<575, 576, 626>,
<627, 626, 576>,
<576, 577, 627>,
<628, 627, 577>,
<577, 578, 628>,
<629, 628, 578>,
<578, 579, 629>,
<630, 629, 579>,
<579, 580, 630>,
<631, 630, 580>,
<580, 581, 631>,
<632, 631, 581>,
<581, 582, 632>,
<633, 632, 582>,
<582, 583, 633>,
<634, 633, 583>,
<583, 584, 634>,
<635, 634, 584>,
<584, 585, 635>,
<636, 635, 585>,
<585, 586, 636>,
<637, 636, 586>,
<586, 587, 637>,
<638, 637, 587>,
<587, 588, 638>,
<639, 638, 588>,
<588, 589, 639>,
<640, 639, 589>,
<589, 590, 640>,
<641, 640, 590>,
<590, 591, 641>,
<642, 641, 591>,
<591, 592, 642>,
<643, 642, 592>,
<592, 593, 643>,
<644, 643, 593>,
<593, 594, 644>,
<645, 644, 594>,
<594, 595, 645>,
<646, 645, 595>,
<595, 596, 646>,
<647, 646, 596>,
<596, 597, 647>,
<648, 647, 597>,
<597, 598, 648>,
<649, 648, 598>,
<598, 599, 649>,
<650, 649, 599>,
<599, 600, 650>,
<651, 650, 600>,
<600, 601, 651>,
<652, 651, 601>,
<601, 602, 652>,
<653, 652, 602>,

<602, 603, 653>,
<654, 653, 603>,
<603, 604, 654>,
<655, 654, 604>,
<604, 605, 655>,
<656, 655, 605>,
<605, 606, 656>,
<657, 656, 606>,
<606, 607, 657>,
<658, 657, 607>,
<607, 608, 658>,
<659, 658, 608>,
<608, 609, 659>,
<660, 659, 609>,
<609, 610, 660>,
<661, 660, 610>,
<610, 611, 661>,
<662, 661, 611>,
<612, 613, 663>,
<664, 663, 613>,
<613, 614, 664>,
<665, 664, 614>,
<614, 615, 665>,
<666, 665, 615>,
<615, 616, 666>,
<667, 666, 616>,
<616, 617, 667>,
<668, 667, 617>,
<617, 618, 668>,
<669, 668, 618>,
<618, 619, 669>,
<670, 669, 619>,
<619, 620, 670>,
<671, 670, 620>,
<620, 621, 671>,
<672, 671, 621>,
<621, 622, 672>,
<673, 672, 622>,
<622, 623, 673>,
<674, 673, 623>,
<623, 624, 674>,
<675, 674, 624>,
<624, 625, 675>,
<676, 675, 625>,
<625, 626, 676>,
<677, 676, 626>,
<626, 627, 677>,
<678, 677, 627>,
<627, 628, 678>,
<679, 678, 628>,
<628, 629, 679>,
<680, 679, 629>,
<629, 630, 680>,
<681, 680, 630>,
<630, 631, 681>,
<682, 681, 631>,

<631, 632, 682>,
<683, 682, 632>,
<632, 633, 683>,
<684, 683, 633>,
<633, 634, 684>,
<685, 684, 634>,
<634, 635, 685>,
<686, 685, 635>,
<635, 636, 686>,
<687, 686, 636>,
<636, 637, 687>,
<688, 687, 637>,
<637, 638, 688>,
<689, 688, 638>,
<638, 639, 689>,
<690, 689, 639>,
<639, 640, 690>,
<691, 690, 640>,
<640, 641, 691>,
<692, 691, 641>,
<641, 642, 692>,
<693, 692, 642>,
<642, 643, 693>,
<694, 693, 643>,
<643, 644, 694>,
<695, 694, 644>,
<644, 645, 695>,
<696, 695, 645>,
<645, 646, 696>,
<697, 696, 646>,
<646, 647, 697>,
<698, 697, 647>,
<647, 648, 698>,
<699, 698, 648>,
<648, 649, 699>,
<700, 699, 649>,
<649, 650, 700>,
<701, 700, 650>,
<650, 651, 701>,
<702, 701, 651>,
<651, 652, 702>,
<703, 702, 652>,
<652, 653, 703>,
<704, 703, 653>,
<653, 654, 704>,
<705, 704, 654>,
<654, 655, 705>,
<706, 705, 655>,
<655, 656, 706>,
<707, 706, 656>,
<656, 657, 707>,
<708, 707, 657>,
<657, 658, 708>,
<709, 708, 658>,
<658, 659, 709>,
<710, 709, 659>,

<659, 660, 710>,
<711, 710, 660>,
<660, 661, 711>,
<712, 711, 661>,
<661, 662, 712>,
<713, 712, 662>,
<663, 664, 714>,
<715, 714, 664>,
<664, 665, 715>,
<716, 715, 665>,
<665, 666, 716>,
<717, 716, 666>,
<666, 667, 717>,
<718, 717, 667>,
<667, 668, 718>,
<719, 718, 668>,
<668, 669, 719>,
<720, 719, 669>,
<669, 670, 720>,
<721, 720, 670>,
<670, 671, 721>,
<722, 721, 671>,
<671, 672, 722>,
<723, 722, 672>,
<672, 673, 723>,
<724, 723, 673>,
<673, 674, 724>,
<725, 724, 674>,
<674, 675, 725>,
<726, 725, 675>,
<675, 676, 726>,
<727, 726, 676>,
<676, 677, 727>,
<728, 727, 677>,
<677, 678, 728>,
<729, 728, 678>,
<678, 679, 729>,
<730, 729, 679>,
<679, 680, 730>,
<731, 730, 680>,
<680, 681, 731>,
<732, 731, 681>,
<681, 682, 732>,
<733, 732, 682>,
<682, 683, 733>,
<734, 733, 683>,
<683, 684, 734>,
<735, 734, 684>,
<684, 685, 735>,
<736, 735, 685>,
<685, 686, 736>,
<737, 736, 686>,
<686, 687, 737>,
<738, 737, 687>,
<687, 688, 738>,
<739, 738, 688>,

<688, 689, 739>,
<740, 739, 689>,
<689, 690, 740>,
<741, 740, 690>,
<690, 691, 741>,
<742, 741, 691>,
<691, 692, 742>,
<743, 742, 692>,
<692, 693, 743>,
<744, 743, 693>,
<693, 694, 744>,
<745, 744, 694>,
<694, 695, 745>,
<746, 745, 695>,
<695, 696, 746>,
<747, 746, 696>,
<696, 697, 747>,
<748, 747, 697>,
<697, 698, 748>,
<749, 748, 698>,
<698, 699, 749>,
<750, 749, 699>,
<699, 700, 750>,
<751, 750, 700>,
<700, 701, 751>,
<752, 751, 701>,
<701, 702, 752>,
<753, 752, 702>,
<702, 703, 753>,
<754, 753, 703>,
<703, 704, 754>,
<755, 754, 704>,
<704, 705, 755>,
<756, 755, 705>,
<705, 706, 756>,
<757, 756, 706>,
<706, 707, 757>,
<758, 757, 707>,
<707, 708, 758>,
<759, 758, 708>,
<708, 709, 759>,
<760, 759, 709>,
<709, 710, 760>,
<761, 760, 710>,
<710, 711, 761>,
<762, 761, 711>,
<711, 712, 762>,
<763, 762, 712>,
<712, 713, 763>,
<764, 763, 713>,
<714, 715, 765>,
<766, 765, 715>,
<715, 716, 766>,
<767, 766, 716>,
<716, 717, 767>,
<768, 767, 717>,

<717,718,768>,
<769,768,718>,
<718,719,769>,
<770,769,719>,
<719,720,770>,
<771,770,720>,
<720,721,771>,
<772,771,721>,
<721,722,772>,
<773,772,722>,
<722,723,773>,
<774,773,723>,
<723,724,774>,
<775,774,724>,
<724,725,775>,
<776,775,725>,
<725,726,776>,
<777,776,726>,
<726,727,777>,
<778,777,727>,
<727,728,778>,
<779,778,728>,
<728,729,779>,
<780,779,729>,
<729,730,780>,
<781,780,730>,
<730,731,781>,
<782,781,731>,
<731,732,782>,
<783,782,732>,
<732,733,783>,
<784,783,733>,
<733,734,784>,
<785,784,734>,
<734,735,785>,
<786,785,735>,
<735,736,786>,
<787,786,736>,
<736,737,787>,
<788,787,737>,
<737,738,788>,
<789,788,738>,
<738,739,789>,
<790,789,739>,
<739,740,790>,
<791,790,740>,
<740,741,791>,
<792,791,741>,
<741,742,792>,
<793,792,742>,
<742,743,793>,
<794,793,743>,
<743,744,794>,
<795,794,744>,
<744,745,795>,
<796,795,745>,

<745, 746, 796>,
<797, 796, 746>,
<746, 747, 797>,
<798, 797, 747>,
<747, 748, 798>,
<799, 798, 748>,
<748, 749, 799>,
<800, 799, 749>,
<749, 750, 800>,
<801, 800, 750>,
<750, 751, 801>,
<802, 801, 751>,
<751, 752, 802>,
<803, 802, 752>,
<752, 753, 803>,
<804, 803, 753>,
<753, 754, 804>,
<805, 804, 754>,
<754, 755, 805>,
<806, 805, 755>,
<755, 756, 806>,
<807, 806, 756>,
<756, 757, 807>,
<808, 807, 757>,
<757, 758, 808>,
<809, 808, 758>,
<758, 759, 809>,
<810, 809, 759>,
<759, 760, 810>,
<811, 810, 760>,
<760, 761, 811>,
<812, 811, 761>,
<761, 762, 812>,
<813, 812, 762>,
<762, 763, 813>,
<814, 813, 763>,
<763, 764, 814>,
<815, 814, 764>,
<765, 766, 816>,
<817, 816, 766>,
<766, 767, 817>,
<818, 817, 767>,
<767, 768, 818>,
<819, 818, 768>,
<768, 769, 819>,
<820, 819, 769>,
<769, 770, 820>,
<821, 820, 770>,
<770, 771, 821>,
<822, 821, 771>,
<771, 772, 822>,
<823, 822, 772>,
<772, 773, 823>,
<824, 823, 773>,
<773, 774, 824>,
<825, 824, 774>,

<774, 775, 825>,
<826, 825, 775>,
<775, 776, 826>,
<827, 826, 776>,
<776, 777, 827>,
<828, 827, 777>,
<777, 778, 828>,
<829, 828, 778>,
<778, 779, 829>,
<830, 829, 779>,
<779, 780, 830>,
<831, 830, 780>,
<780, 781, 831>,
<832, 831, 781>,
<781, 782, 832>,
<833, 832, 782>,
<782, 783, 833>,
<834, 833, 783>,
<783, 784, 834>,
<835, 834, 784>,
<784, 785, 835>,
<836, 835, 785>,
<785, 786, 836>,
<837, 836, 786>,
<786, 787, 837>,
<838, 837, 787>,
<787, 788, 838>,
<839, 838, 788>,
<788, 789, 839>,
<840, 839, 789>,
<789, 790, 840>,
<841, 840, 790>,
<790, 791, 841>,
<842, 841, 791>,
<791, 792, 842>,
<843, 842, 792>,
<792, 793, 843>,
<844, 843, 793>,
<793, 794, 844>,
<845, 844, 794>,
<794, 795, 845>,
<846, 845, 795>,
<795, 796, 846>,
<847, 846, 796>,
<796, 797, 847>,
<848, 847, 797>,
<797, 798, 848>,
<849, 848, 798>,
<798, 799, 849>,
<850, 849, 799>,
<799, 800, 850>,
<851, 850, 800>,
<800, 801, 851>,
<852, 851, 801>,
<801, 802, 852>,
<853, 852, 802>,

<802,803,853>,
<854,853,803>,
<803,804,854>,
<855,854,804>,
<804,805,855>,
<856,855,805>,
<805,806,856>,
<857,856,806>,
<806,807,857>,
<858,857,807>,
<807,808,858>,
<859,858,808>,
<808,809,859>,
<860,859,809>,
<809,810,860>,
<861,860,810>,
<810,811,861>,
<862,861,811>,
<811,812,862>,
<863,862,812>,
<812,813,863>,
<864,863,813>,
<813,814,864>,
<865,864,814>,
<814,815,865>,
<866,865,815>,
<816,817,867>,
<868,867,817>,
<817,818,868>,
<869,868,818>,
<818,819,869>,
<870,869,819>,
<819,820,870>,
<871,870,820>,
<820,821,871>,
<872,871,821>,
<821,822,872>,
<873,872,822>,
<822,823,873>,
<874,873,823>,
<823,824,874>,
<875,874,824>,
<824,825,875>,
<876,875,825>,
<825,826,876>,
<877,876,826>,
<826,827,877>,
<878,877,827>,
<827,828,878>,
<879,878,828>,
<828,829,879>,
<880,879,829>,
<829,830,880>,
<881,880,830>,
<830,831,881>,
<882,881,831>,

<831, 832, 882>,
<883, 882, 832>,
<832, 833, 883>,
<884, 883, 833>,
<833, 834, 884>,
<885, 884, 834>,
<834, 835, 885>,
<886, 885, 835>,
<835, 836, 886>,
<887, 886, 836>,
<836, 837, 887>,
<888, 887, 837>,
<837, 838, 888>,
<889, 888, 838>,
<838, 839, 889>,
<890, 889, 839>,
<839, 840, 890>,
<891, 890, 840>,
<840, 841, 891>,
<892, 891, 841>,
<841, 842, 892>,
<893, 892, 842>,
<842, 843, 893>,
<894, 893, 843>,
<843, 844, 894>,
<895, 894, 844>,
<844, 845, 895>,
<896, 895, 845>,
<845, 846, 896>,
<897, 896, 846>,
<846, 847, 897>,
<898, 897, 847>,
<847, 848, 898>,
<899, 898, 848>,
<848, 849, 899>,
<900, 899, 849>,
<849, 850, 900>,
<901, 900, 850>,
<850, 851, 901>,
<902, 901, 851>,
<851, 852, 902>,
<903, 902, 852>,
<852, 853, 903>,
<904, 903, 853>,
<853, 854, 904>,
<905, 904, 854>,
<854, 855, 905>,
<906, 905, 855>,
<855, 856, 906>,
<907, 906, 856>,
<856, 857, 907>,
<908, 907, 857>,
<857, 858, 908>,
<909, 908, 858>,
<858, 859, 909>,
<910, 909, 859>,

<859, 860, 910>,
<911, 910, 860>,
<860, 861, 911>,
<912, 911, 861>,
<861, 862, 912>,
<913, 912, 862>,
<862, 863, 913>,
<914, 913, 863>,
<863, 864, 914>,
<915, 914, 864>,
<864, 865, 915>,
<916, 915, 865>,
<865, 866, 916>,
<917, 916, 866>,
<867, 868, 918>,
<919, 918, 868>,
<868, 869, 919>,
<920, 919, 869>,
<869, 870, 920>,
<921, 920, 870>,
<870, 871, 921>,
<922, 921, 871>,
<871, 872, 922>,
<923, 922, 872>,
<872, 873, 923>,
<924, 923, 873>,
<873, 874, 924>,
<925, 924, 874>,
<874, 875, 925>,
<926, 925, 875>,
<875, 876, 926>,
<927, 926, 876>,
<876, 877, 927>,
<928, 927, 877>,
<877, 878, 928>,
<929, 928, 878>,
<878, 879, 929>,
<930, 929, 879>,
<879, 880, 930>,
<931, 930, 880>,
<880, 881, 931>,
<932, 931, 881>,
<881, 882, 932>,
<933, 932, 882>,
<882, 883, 933>,
<934, 933, 883>,
<883, 884, 934>,
<935, 934, 884>,
<884, 885, 935>,
<936, 935, 885>,
<885, 886, 936>,
<937, 936, 886>,
<886, 887, 937>,
<938, 937, 887>,
<887, 888, 938>,
<939, 938, 888>,

<888, 889, 939>,
<940, 939, 889>,
<889, 890, 940>,
<941, 940, 890>,
<890, 891, 941>,
<942, 941, 891>,
<891, 892, 942>,
<943, 942, 892>,
<892, 893, 943>,
<944, 943, 893>,
<893, 894, 944>,
<945, 944, 894>,
<894, 895, 945>,
<946, 945, 895>,
<895, 896, 946>,
<947, 946, 896>,
<896, 897, 947>,
<948, 947, 897>,
<897, 898, 948>,
<949, 948, 898>,
<898, 899, 949>,
<950, 949, 899>,
<899, 900, 950>,
<951, 950, 900>,
<900, 901, 951>,
<952, 951, 901>,
<901, 902, 952>,
<953, 952, 902>,
<902, 903, 953>,
<954, 953, 903>,
<903, 904, 954>,
<955, 954, 904>,
<904, 905, 955>,
<956, 955, 905>,
<905, 906, 956>,
<957, 956, 906>,
<906, 907, 957>,
<958, 957, 907>,
<907, 908, 958>,
<959, 958, 908>,
<908, 909, 959>,
<960, 959, 909>,
<909, 910, 960>,
<961, 960, 910>,
<910, 911, 961>,
<962, 961, 911>,
<911, 912, 962>,
<963, 962, 912>,
<912, 913, 963>,
<964, 963, 913>,
<913, 914, 964>,
<965, 964, 914>,
<914, 915, 965>,
<966, 965, 915>,
<915, 916, 966>,
<967, 966, 916>,

<916, 917, 967>,
<968, 967, 917>,
<918, 919, 969>,
<970, 969, 919>,
<919, 920, 970>,
<971, 970, 920>,
<920, 921, 971>,
<972, 971, 921>,
<921, 922, 972>,
<973, 972, 922>,
<922, 923, 973>,
<974, 973, 923>,
<923, 924, 974>,
<975, 974, 924>,
<924, 925, 975>,
<976, 975, 925>,
<925, 926, 976>,
<977, 976, 926>,
<926, 927, 977>,
<978, 977, 927>,
<927, 928, 978>,
<979, 978, 928>,
<928, 929, 979>,
<980, 979, 929>,
<929, 930, 980>,
<981, 980, 930>,
<930, 931, 981>,
<982, 981, 931>,
<931, 932, 982>,
<983, 982, 932>,
<932, 933, 983>,
<984, 983, 933>,
<933, 934, 984>,
<985, 984, 934>,
<934, 935, 985>,
<986, 985, 935>,
<935, 936, 986>,
<987, 986, 936>,
<936, 937, 987>,
<988, 987, 937>,
<937, 938, 988>,
<989, 988, 938>,
<938, 939, 989>,
<990, 989, 939>,
<939, 940, 990>,
<991, 990, 940>,
<940, 941, 991>,
<992, 991, 941>,
<941, 942, 992>,
<993, 992, 942>,
<942, 943, 993>,
<994, 993, 943>,
<943, 944, 994>,
<995, 994, 944>,
<944, 945, 995>,
<996, 995, 945>,

<945, 946, 996>,
<997, 996, 946>,
<946, 947, 997>,
<998, 997, 947>,
<947, 948, 998>,
<999, 998, 948>,
<948, 949, 999>,
<1000, 999, 949>,
<949, 950, 1000>,
<1001, 1000, 950>,
<950, 951, 1001>,
<1002, 1001, 951>,
<951, 952, 1002>,
<1003, 1002, 952>,
<952, 953, 1003>,
<1004, 1003, 953>,
<953, 954, 1004>,
<1005, 1004, 954>,
<954, 955, 1005>,
<1006, 1005, 955>,
<955, 956, 1006>,
<1007, 1006, 956>,
<956, 957, 1007>,
<1008, 1007, 957>,
<957, 958, 1008>,
<1009, 1008, 958>,
<958, 959, 1009>,
<1010, 1009, 959>,
<959, 960, 1010>,
<1011, 1010, 960>,
<960, 961, 1011>,
<1012, 1011, 961>,
<961, 962, 1012>,
<1013, 1012, 962>,
<962, 963, 1013>,
<1014, 1013, 963>,
<963, 964, 1014>,
<1015, 1014, 964>,
<964, 965, 1015>,
<1016, 1015, 965>,
<965, 966, 1016>,
<1017, 1016, 966>,
<966, 967, 1017>,
<1018, 1017, 967>,
<967, 968, 1018>,
<1019, 1018, 968>,
<969, 970, 1020>,
<1021, 1020, 970>,
<970, 971, 1021>,
<1022, 1021, 971>,
<971, 972, 1022>,
<1023, 1022, 972>,
<972, 973, 1023>,
<1024, 1023, 973>,
<973, 974, 1024>,
<1025, 1024, 974>,

<974, 975, 1025>,
<1026, 1025, 975>,
<975, 976, 1026>,
<1027, 1026, 976>,
<976, 977, 1027>,
<1028, 1027, 977>,
<977, 978, 1028>,
<1029, 1028, 978>,
<978, 979, 1029>,
<1030, 1029, 979>,
<979, 980, 1030>,
<1031, 1030, 980>,
<980, 981, 1031>,
<1032, 1031, 981>,
<981, 982, 1032>,
<1033, 1032, 982>,
<982, 983, 1033>,
<1034, 1033, 983>,
<983, 984, 1034>,
<1035, 1034, 984>,
<984, 985, 1035>,
<1036, 1035, 985>,
<985, 986, 1036>,
<1037, 1036, 986>,
<986, 987, 1037>,
<1038, 1037, 987>,
<987, 988, 1038>,
<1039, 1038, 988>,
<988, 989, 1039>,
<1040, 1039, 989>,
<989, 990, 1040>,
<1041, 1040, 990>,
<990, 991, 1041>,
<1042, 1041, 991>,
<991, 992, 1042>,
<1043, 1042, 992>,
<992, 993, 1043>,
<1044, 1043, 993>,
<993, 994, 1044>,
<1045, 1044, 994>,
<994, 995, 1045>,
<1046, 1045, 995>,
<995, 996, 1046>,
<1047, 1046, 996>,
<996, 997, 1047>,
<1048, 1047, 997>,
<997, 998, 1048>,
<1049, 1048, 998>,
<998, 999, 1049>,
<1050, 1049, 999>,
<999, 1000, 1050>,
<1051, 1050, 1000>,
<1000, 1001, 1051>,
<1052, 1051, 1001>,
<1001, 1002, 1052>,
<1053, 1052, 1002>,

<1002,1003,1053>,
<1054,1053,1003>,
<1003,1004,1054>,
<1055,1054,1004>,
<1004,1005,1055>,
<1056,1055,1005>,
<1005,1006,1056>,
<1057,1056,1006>,
<1006,1007,1057>,
<1058,1057,1007>,
<1007,1008,1058>,
<1059,1058,1008>,
<1008,1009,1059>,
<1060,1059,1009>,
<1009,1010,1060>,
<1061,1060,1010>,
<1010,1011,1061>,
<1062,1061,1011>,
<1011,1012,1062>,
<1063,1062,1012>,
<1012,1013,1063>,
<1064,1063,1013>,
<1013,1014,1064>,
<1065,1064,1014>,
<1014,1015,1065>,
<1066,1065,1015>,
<1015,1016,1066>,
<1067,1066,1016>,
<1016,1017,1067>,
<1068,1067,1017>,
<1017,1018,1068>,
<1069,1068,1018>,
<1018,1019,1069>,
<1070,1069,1019>,
<1020,1021,1071>,
<1072,1071,1021>,
<1021,1022,1072>,
<1073,1072,1022>,
<1022,1023,1073>,
<1074,1073,1023>,
<1023,1024,1074>,
<1075,1074,1024>,
<1024,1025,1075>,
<1076,1075,1025>,
<1025,1026,1076>,
<1077,1076,1026>,
<1026,1027,1077>,
<1078,1077,1027>,
<1027,1028,1078>,
<1079,1078,1028>,
<1028,1029,1079>,
<1080,1079,1029>,
<1029,1030,1080>,
<1081,1080,1030>,
<1030,1031,1081>,
<1082,1081,1031>,

<1031,1032,1082>,
<1083,1082,1032>,
<1032,1033,1083>,
<1084,1083,1033>,
<1033,1034,1084>,
<1085,1084,1034>,
<1034,1035,1085>,
<1086,1085,1035>,
<1035,1036,1086>,
<1087,1086,1036>,
<1036,1037,1087>,
<1088,1087,1037>,
<1037,1038,1088>,
<1089,1088,1038>,
<1038,1039,1089>,
<1090,1089,1039>,
<1039,1040,1090>,
<1091,1090,1040>,
<1040,1041,1091>,
<1092,1091,1041>,
<1041,1042,1092>,
<1093,1092,1042>,
<1042,1043,1093>,
<1094,1093,1043>,
<1043,1044,1094>,
<1095,1094,1044>,
<1044,1045,1095>,
<1096,1095,1045>,
<1045,1046,1096>,
<1097,1096,1046>,
<1046,1047,1097>,
<1098,1097,1047>,
<1047,1048,1098>,
<1099,1098,1048>,
<1048,1049,1099>,
<1100,1099,1049>,
<1049,1050,1100>,
<1101,1100,1050>,
<1050,1051,1101>,
<1102,1101,1051>,
<1051,1052,1102>,
<1103,1102,1052>,
<1052,1053,1103>,
<1104,1103,1053>,
<1053,1054,1104>,
<1105,1104,1054>,
<1054,1055,1105>,
<1106,1105,1055>,
<1055,1056,1106>,
<1107,1106,1056>,
<1056,1057,1107>,
<1108,1107,1057>,
<1057,1058,1108>,
<1109,1108,1058>,
<1058,1059,1109>,
<1110,1109,1059>,

<1059,1060,1110>,
<1111,1110,1060>,
<1060,1061,1111>,
<1112,1111,1061>,
<1061,1062,1112>,
<1113,1112,1062>,
<1062,1063,1113>,
<1114,1113,1063>,
<1063,1064,1114>,
<1115,1114,1064>,
<1064,1065,1115>,
<1116,1115,1065>,
<1065,1066,1116>,
<1117,1116,1066>,
<1066,1067,1117>,
<1118,1117,1067>,
<1067,1068,1118>,
<1119,1118,1068>,
<1068,1069,1119>,
<1120,1119,1069>,
<1069,1070,1120>,
<1121,1120,1070>,
<1071,1072,1122>,
<1123,1122,1072>,
<1072,1073,1123>,
<1124,1123,1073>,
<1073,1074,1124>,
<1125,1124,1074>,
<1074,1075,1125>,
<1126,1125,1075>,
<1075,1076,1126>,
<1127,1126,1076>,
<1076,1077,1127>,
<1128,1127,1077>,
<1077,1078,1128>,
<1129,1128,1078>,
<1078,1079,1129>,
<1130,1129,1079>,
<1079,1080,1130>,
<1131,1130,1080>,
<1080,1081,1131>,
<1132,1131,1081>,
<1081,1082,1132>,
<1133,1132,1082>,
<1082,1083,1133>,
<1134,1133,1083>,
<1083,1084,1134>,
<1135,1134,1084>,
<1084,1085,1135>,
<1136,1135,1085>,
<1085,1086,1136>,
<1137,1136,1086>,
<1086,1087,1137>,
<1138,1137,1087>,
<1087,1088,1138>,
<1139,1138,1088>,

<1088,1089,1139>,
<1140,1139,1089>,
<1089,1090,1140>,
<1141,1140,1090>,
<1090,1091,1141>,
<1142,1141,1091>,
<1091,1092,1142>,
<1143,1142,1092>,
<1092,1093,1143>,
<1144,1143,1093>,
<1093,1094,1144>,
<1145,1144,1094>,
<1094,1095,1145>,
<1146,1145,1095>,
<1095,1096,1146>,
<1147,1146,1096>,
<1096,1097,1147>,
<1148,1147,1097>,
<1097,1098,1148>,
<1149,1148,1098>,
<1098,1099,1149>,
<1150,1149,1099>,
<1099,1100,1150>,
<1151,1150,1100>,
<1100,1101,1151>,
<1152,1151,1101>,
<1101,1102,1152>,
<1153,1152,1102>,
<1102,1103,1153>,
<1154,1153,1103>,
<1103,1104,1154>,
<1155,1154,1104>,
<1104,1105,1155>,
<1156,1155,1105>,
<1105,1106,1156>,
<1157,1156,1106>,
<1106,1107,1157>,
<1158,1157,1107>,
<1107,1108,1158>,
<1159,1158,1108>,
<1108,1109,1159>,
<1160,1159,1109>,
<1109,1110,1160>,
<1161,1160,1110>,
<1110,1111,1161>,
<1162,1161,1111>,
<1111,1112,1162>,
<1163,1162,1112>,
<1112,1113,1163>,
<1164,1163,1113>,
<1113,1114,1164>,
<1165,1164,1114>,
<1114,1115,1165>,
<1166,1165,1115>,
<1115,1116,1166>,
<1167,1166,1116>,

<1116,1117,1167>,
<1168,1167,1117>,
<1117,1118,1168>,
<1169,1168,1118>,
<1118,1119,1169>,
<1170,1169,1119>,
<1119,1120,1170>,
<1171,1170,1120>,
<1120,1121,1171>,
<1172,1171,1121>,
<1122,1123,1173>,
<1174,1173,1123>,
<1123,1124,1174>,
<1175,1174,1124>,
<1124,1125,1175>,
<1176,1175,1125>,
<1125,1126,1176>,
<1177,1176,1126>,
<1126,1127,1177>,
<1178,1177,1127>,
<1127,1128,1178>,
<1179,1178,1128>,
<1128,1129,1179>,
<1180,1179,1129>,
<1129,1130,1180>,
<1181,1180,1130>,
<1130,1131,1181>,
<1182,1181,1131>,
<1131,1132,1182>,
<1183,1182,1132>,
<1132,1133,1183>,
<1184,1183,1133>,
<1133,1134,1184>,
<1185,1184,1134>,
<1134,1135,1185>,
<1186,1185,1135>,
<1135,1136,1186>,
<1187,1186,1136>,
<1136,1137,1187>,
<1188,1187,1137>,
<1137,1138,1188>,
<1189,1188,1138>,
<1138,1139,1189>,
<1190,1189,1139>,
<1139,1140,1190>,
<1191,1190,1140>,
<1140,1141,1191>,
<1192,1191,1141>,
<1141,1142,1192>,
<1193,1192,1142>,
<1142,1143,1193>,
<1194,1193,1143>,
<1143,1144,1194>,
<1195,1194,1144>,
<1144,1145,1195>,
<1196,1195,1145>,

<1145,1146,1196>,
<1197,1196,1146>,
<1146,1147,1197>,
<1198,1197,1147>,
<1147,1148,1198>,
<1199,1198,1148>,
<1148,1149,1199>,
<1200,1199,1149>,
<1149,1150,1200>,
<1201,1200,1150>,
<1150,1151,1201>,
<1202,1201,1151>,
<1151,1152,1202>,
<1203,1202,1152>,
<1152,1153,1203>,
<1204,1203,1153>,
<1153,1154,1204>,
<1205,1204,1154>,
<1154,1155,1205>,
<1206,1205,1155>,
<1155,1156,1206>,
<1207,1206,1156>,
<1156,1157,1207>,
<1208,1207,1157>,
<1157,1158,1208>,
<1209,1208,1158>,
<1158,1159,1209>,
<1210,1209,1159>,
<1159,1160,1210>,
<1211,1210,1160>,
<1160,1161,1211>,
<1212,1211,1161>,
<1161,1162,1212>,
<1213,1212,1162>,
<1162,1163,1213>,
<1214,1213,1163>,
<1163,1164,1214>,
<1215,1214,1164>,
<1164,1165,1215>,
<1216,1215,1165>,
<1165,1166,1216>,
<1217,1216,1166>,
<1166,1167,1217>,
<1218,1217,1167>,
<1167,1168,1218>,
<1219,1218,1168>,
<1168,1169,1219>,
<1220,1219,1169>,
<1169,1170,1220>,
<1221,1220,1170>,
<1170,1171,1221>,
<1222,1221,1171>,
<1171,1172,1222>,
<1223,1222,1172>,
<1173,1174,1224>,
<1225,1224,1174>,

<1174,1175,1225>,
<1226,1225,1175>,
<1175,1176,1226>,
<1227,1226,1176>,
<1176,1177,1227>,
<1228,1227,1177>,
<1177,1178,1228>,
<1229,1228,1178>,
<1178,1179,1229>,
<1230,1229,1179>,
<1179,1180,1230>,
<1231,1230,1180>,
<1180,1181,1231>,
<1232,1231,1181>,
<1181,1182,1232>,
<1233,1232,1182>,
<1182,1183,1233>,
<1234,1233,1183>,
<1183,1184,1234>,
<1235,1234,1184>,
<1184,1185,1235>,
<1236,1235,1185>,
<1185,1186,1236>,
<1237,1236,1186>,
<1186,1187,1237>,
<1238,1237,1187>,
<1187,1188,1238>,
<1239,1238,1188>,
<1188,1189,1239>,
<1240,1239,1189>,
<1189,1190,1240>,
<1241,1240,1190>,
<1190,1191,1241>,
<1242,1241,1191>,
<1191,1192,1242>,
<1243,1242,1192>,
<1192,1193,1243>,
<1244,1243,1193>,
<1193,1194,1244>,
<1245,1244,1194>,
<1194,1195,1245>,
<1246,1245,1195>,
<1195,1196,1246>,
<1247,1246,1196>,
<1196,1197,1247>,
<1248,1247,1197>,
<1197,1198,1248>,
<1249,1248,1198>,
<1198,1199,1249>,
<1250,1249,1199>,
<1199,1200,1250>,
<1251,1250,1200>,
<1200,1201,1251>,
<1252,1251,1201>,
<1201,1202,1252>,
<1253,1252,1202>,

<1202,1203,1253>,
<1254,1253,1203>,
<1203,1204,1254>,
<1255,1254,1204>,
<1204,1205,1255>,
<1256,1255,1205>,
<1205,1206,1256>,
<1257,1256,1206>,
<1206,1207,1257>,
<1258,1257,1207>,
<1207,1208,1258>,
<1259,1258,1208>,
<1208,1209,1259>,
<1260,1259,1209>,
<1209,1210,1260>,
<1261,1260,1210>,
<1210,1211,1261>,
<1262,1261,1211>,
<1211,1212,1262>,
<1263,1262,1212>,
<1212,1213,1263>,
<1264,1263,1213>,
<1213,1214,1264>,
<1265,1264,1214>,
<1214,1215,1265>,
<1266,1265,1215>,
<1215,1216,1266>,
<1267,1266,1216>,
<1216,1217,1267>,
<1268,1267,1217>,
<1217,1218,1268>,
<1269,1268,1218>,
<1218,1219,1269>,
<1270,1269,1219>,
<1219,1220,1270>,
<1271,1270,1220>,
<1220,1221,1271>,
<1272,1271,1221>,
<1221,1222,1272>,
<1273,1272,1222>,
<1222,1223,1273>,
<1274,1273,1223>,
<1224,1225,1275>,
<1276,1275,1225>,
<1225,1226,1276>,
<1277,1276,1226>,
<1226,1227,1277>,
<1278,1277,1227>,
<1227,1228,1278>,
<1279,1278,1228>,
<1228,1229,1279>,
<1280,1279,1229>,
<1229,1230,1280>,
<1281,1280,1230>,
<1230,1231,1281>,
<1282,1281,1231>,

<1231,1232,1282>,
<1283,1282,1232>,
<1232,1233,1283>,
<1284,1283,1233>,
<1233,1234,1284>,
<1285,1284,1234>,
<1234,1235,1285>,
<1286,1285,1235>,
<1235,1236,1286>,
<1287,1286,1236>,
<1236,1237,1287>,
<1288,1287,1237>,
<1237,1238,1288>,
<1289,1288,1238>,
<1238,1239,1289>,
<1290,1289,1239>,
<1239,1240,1290>,
<1291,1290,1240>,
<1240,1241,1291>,
<1292,1291,1241>,
<1241,1242,1292>,
<1293,1292,1242>,
<1242,1243,1293>,
<1294,1293,1243>,
<1243,1244,1294>,
<1295,1294,1244>,
<1244,1245,1295>,
<1296,1295,1245>,
<1245,1246,1296>,
<1297,1296,1246>,
<1246,1247,1297>,
<1298,1297,1247>,
<1247,1248,1298>,
<1299,1298,1248>,
<1248,1249,1299>,
<1300,1299,1249>,
<1249,1250,1300>,
<1301,1300,1250>,
<1250,1251,1301>,
<1302,1301,1251>,
<1251,1252,1302>,
<1303,1302,1252>,
<1252,1253,1303>,
<1304,1303,1253>,
<1253,1254,1304>,
<1305,1304,1254>,
<1254,1255,1305>,
<1306,1305,1255>,
<1255,1256,1306>,
<1307,1306,1256>,
<1256,1257,1307>,
<1308,1307,1257>,
<1257,1258,1308>,
<1309,1308,1258>,
<1258,1259,1309>,
<1310,1309,1259>,

<1259,1260,1310>,
<1311,1310,1260>,
<1260,1261,1311>,
<1312,1311,1261>,
<1261,1262,1312>,
<1313,1312,1262>,
<1262,1263,1313>,
<1314,1313,1263>,
<1263,1264,1314>,
<1315,1314,1264>,
<1264,1265,1315>,
<1316,1315,1265>,
<1265,1266,1316>,
<1317,1316,1266>,
<1266,1267,1317>,
<1318,1317,1267>,
<1267,1268,1318>,
<1319,1318,1268>,
<1268,1269,1319>,
<1320,1319,1269>,
<1269,1270,1320>,
<1321,1320,1270>,
<1270,1271,1321>,
<1322,1321,1271>,
<1271,1272,1322>,
<1323,1322,1272>,
<1272,1273,1323>,
<1324,1323,1273>,
<1273,1274,1324>,
<1325,1324,1274>,
<1275,1276,1326>,
<1327,1326,1276>,
<1276,1277,1327>,
<1328,1327,1277>,
<1277,1278,1328>,
<1329,1328,1278>,
<1278,1279,1329>,
<1330,1329,1279>,
<1279,1280,1330>,
<1331,1330,1280>,
<1280,1281,1331>,
<1332,1331,1281>,
<1281,1282,1332>,
<1333,1332,1282>,
<1282,1283,1333>,
<1334,1333,1283>,
<1283,1284,1334>,
<1335,1334,1284>,
<1284,1285,1335>,
<1336,1335,1285>,
<1285,1286,1336>,
<1337,1336,1286>,
<1286,1287,1337>,
<1338,1337,1287>,
<1287,1288,1338>,
<1339,1338,1288>,

<1288,1289,1339>,
<1340,1339,1289>,
<1289,1290,1340>,
<1341,1340,1290>,
<1290,1291,1341>,
<1342,1341,1291>,
<1291,1292,1342>,
<1343,1342,1292>,
<1292,1293,1343>,
<1344,1343,1293>,
<1293,1294,1344>,
<1345,1344,1294>,
<1294,1295,1345>,
<1346,1345,1295>,
<1295,1296,1346>,
<1347,1346,1296>,
<1296,1297,1347>,
<1348,1347,1297>,
<1297,1298,1348>,
<1349,1348,1298>,
<1298,1299,1349>,
<1350,1349,1299>,
<1299,1300,1350>,
<1351,1350,1300>,
<1300,1301,1351>,
<1352,1351,1301>,
<1301,1302,1352>,
<1353,1352,1302>,
<1302,1303,1353>,
<1354,1353,1303>,
<1303,1304,1354>,
<1355,1354,1304>,
<1304,1305,1355>,
<1356,1355,1305>,
<1305,1306,1356>,
<1357,1356,1306>,
<1306,1307,1357>,
<1358,1357,1307>,
<1307,1308,1358>,
<1359,1358,1308>,
<1308,1309,1359>,
<1360,1359,1309>,
<1309,1310,1360>,
<1361,1360,1310>,
<1310,1311,1361>,
<1362,1361,1311>,
<1311,1312,1362>,
<1363,1362,1312>,
<1312,1313,1363>,
<1364,1363,1313>,
<1313,1314,1364>,
<1365,1364,1314>,
<1314,1315,1365>,
<1366,1365,1315>,
<1315,1316,1366>,
<1367,1366,1316>,

<1316,1317,1367>,
<1368,1367,1317>,
<1317,1318,1368>,
<1369,1368,1318>,
<1318,1319,1369>,
<1370,1369,1319>,
<1319,1320,1370>,
<1371,1370,1320>,
<1320,1321,1371>,
<1372,1371,1321>,
<1321,1322,1372>,
<1373,1372,1322>,
<1322,1323,1373>,
<1374,1373,1323>,
<1323,1324,1374>,
<1375,1374,1324>,
<1324,1325,1375>,
<1376,1375,1325>,
<1326,1327,1377>,
<1378,1377,1327>,
<1327,1328,1378>,
<1379,1378,1328>,
<1328,1329,1379>,
<1380,1379,1329>,
<1329,1330,1380>,
<1381,1380,1330>,
<1330,1331,1381>,
<1382,1381,1331>,
<1331,1332,1382>,
<1383,1382,1332>,
<1332,1333,1383>,
<1384,1383,1333>,
<1333,1334,1384>,
<1385,1384,1334>,
<1334,1335,1385>,
<1386,1385,1335>,
<1335,1336,1386>,
<1387,1386,1336>,
<1336,1337,1387>,
<1388,1387,1337>,
<1337,1338,1388>,
<1389,1388,1338>,
<1338,1339,1389>,
<1390,1389,1339>,
<1339,1340,1390>,
<1391,1390,1340>,
<1340,1341,1391>,
<1392,1391,1341>,
<1341,1342,1392>,
<1393,1392,1342>,
<1342,1343,1393>,
<1394,1393,1343>,
<1343,1344,1394>,
<1395,1394,1344>,
<1344,1345,1395>,
<1396,1395,1345>,

<1345,1346,1396>,
<1397,1396,1346>,
<1346,1347,1397>,
<1398,1397,1347>,
<1347,1348,1398>,
<1399,1398,1348>,
<1348,1349,1399>,
<1400,1399,1349>,
<1349,1350,1400>,
<1401,1400,1350>,
<1350,1351,1401>,
<1402,1401,1351>,
<1351,1352,1402>,
<1403,1402,1352>,
<1352,1353,1403>,
<1404,1403,1353>,
<1353,1354,1404>,
<1405,1404,1354>,
<1354,1355,1405>,
<1406,1405,1355>,
<1355,1356,1406>,
<1407,1406,1356>,
<1356,1357,1407>,
<1408,1407,1357>,
<1357,1358,1408>,
<1409,1408,1358>,
<1358,1359,1409>,
<1410,1409,1359>,
<1359,1360,1410>,
<1411,1410,1360>,
<1360,1361,1411>,
<1412,1411,1361>,
<1361,1362,1412>,
<1413,1412,1362>,
<1362,1363,1413>,
<1414,1413,1363>,
<1363,1364,1414>,
<1415,1414,1364>,
<1364,1365,1415>,
<1416,1415,1365>,
<1365,1366,1416>,
<1417,1416,1366>,
<1366,1367,1417>,
<1418,1417,1367>,
<1367,1368,1418>,
<1419,1418,1368>,
<1368,1369,1419>,
<1420,1419,1369>,
<1369,1370,1420>,
<1421,1420,1370>,
<1370,1371,1421>,
<1422,1421,1371>,
<1371,1372,1422>,
<1423,1422,1372>,
<1372,1373,1423>,
<1424,1423,1373>,

<1373,1374,1424>,
<1425,1424,1374>,
<1374,1375,1425>,
<1426,1425,1375>,
<1375,1376,1426>,
<1427,1426,1376>,
<1377,1378,1428>,
<1429,1428,1378>,
<1378,1379,1429>,
<1430,1429,1379>,
<1379,1380,1430>,
<1431,1430,1380>,
<1380,1381,1431>,
<1432,1431,1381>,
<1381,1382,1432>,
<1433,1432,1382>,
<1382,1383,1433>,
<1434,1433,1383>,
<1383,1384,1434>,
<1435,1434,1384>,
<1384,1385,1435>,
<1436,1435,1385>,
<1385,1386,1436>,
<1437,1436,1386>,
<1386,1387,1437>,
<1438,1437,1387>,
<1387,1388,1438>,
<1439,1438,1388>,
<1388,1389,1439>,
<1440,1439,1389>,
<1389,1390,1440>,
<1441,1440,1390>,
<1390,1391,1441>,
<1442,1441,1391>,
<1391,1392,1442>,
<1443,1442,1392>,
<1392,1393,1443>,
<1444,1443,1393>,
<1393,1394,1444>,
<1445,1444,1394>,
<1394,1395,1445>,
<1446,1445,1395>,
<1395,1396,1446>,
<1447,1446,1396>,
<1396,1397,1447>,
<1448,1447,1397>,
<1397,1398,1448>,
<1449,1448,1398>,
<1398,1399,1449>,
<1450,1449,1399>,
<1399,1400,1450>,
<1451,1450,1400>,
<1400,1401,1451>,
<1452,1451,1401>,
<1401,1402,1452>,
<1453,1452,1402>,

<1402,1403,1453>,
<1454,1453,1403>,
<1403,1404,1454>,
<1455,1454,1404>,
<1404,1405,1455>,
<1456,1455,1405>,
<1405,1406,1456>,
<1457,1456,1406>,
<1406,1407,1457>,
<1458,1457,1407>,
<1407,1408,1458>,
<1459,1458,1408>,
<1408,1409,1459>,
<1460,1459,1409>,
<1409,1410,1460>,
<1461,1460,1410>,
<1410,1411,1461>,
<1462,1461,1411>,
<1411,1412,1462>,
<1463,1462,1412>,
<1412,1413,1463>,
<1464,1463,1413>,
<1413,1414,1464>,
<1465,1464,1414>,
<1414,1415,1465>,
<1466,1465,1415>,
<1415,1416,1466>,
<1467,1466,1416>,
<1416,1417,1467>,
<1468,1467,1417>,
<1417,1418,1468>,
<1469,1468,1418>,
<1418,1419,1469>,
<1470,1469,1419>,
<1419,1420,1470>,
<1471,1470,1420>,
<1420,1421,1471>,
<1472,1471,1421>,
<1421,1422,1472>,
<1473,1472,1422>,
<1422,1423,1473>,
<1474,1473,1423>,
<1423,1424,1474>,
<1475,1474,1424>,
<1424,1425,1475>,
<1476,1475,1425>,
<1425,1426,1476>,
<1477,1476,1426>,
<1426,1427,1477>,
<1478,1477,1427>,
<1428,1429,1479>,
<1480,1479,1429>,
<1429,1430,1480>,
<1481,1480,1430>,
<1430,1431,1481>,
<1482,1481,1431>,

<1431,1432,1482>,
<1483,1482,1432>,
<1432,1433,1483>,
<1484,1483,1433>,
<1433,1434,1484>,
<1485,1484,1434>,
<1434,1435,1485>,
<1486,1485,1435>,
<1435,1436,1486>,
<1487,1486,1436>,
<1436,1437,1487>,
<1488,1487,1437>,
<1437,1438,1488>,
<1489,1488,1438>,
<1438,1439,1489>,
<1490,1489,1439>,
<1439,1440,1490>,
<1491,1490,1440>,
<1440,1441,1491>,
<1492,1491,1441>,
<1441,1442,1492>,
<1493,1492,1442>,
<1442,1443,1493>,
<1494,1493,1443>,
<1443,1444,1494>,
<1495,1494,1444>,
<1444,1445,1495>,
<1496,1495,1445>,
<1445,1446,1496>,
<1497,1496,1446>,
<1446,1447,1497>,
<1498,1497,1447>,
<1447,1448,1498>,
<1499,1498,1448>,
<1448,1449,1499>,
<1500,1499,1449>,
<1449,1450,1500>,
<1501,1500,1450>,
<1450,1451,1501>,
<1502,1501,1451>,
<1451,1452,1502>,
<1503,1502,1452>,
<1452,1453,1503>,
<1504,1503,1453>,
<1453,1454,1504>,
<1505,1504,1454>,
<1454,1455,1505>,
<1506,1505,1455>,
<1455,1456,1506>,
<1507,1506,1456>,
<1456,1457,1507>,
<1508,1507,1457>,
<1457,1458,1508>,
<1509,1508,1458>,
<1458,1459,1509>,
<1510,1509,1459>,

<1459,1460,1510>,
<1511,1510,1460>,
<1460,1461,1511>,
<1512,1511,1461>,
<1461,1462,1512>,
<1513,1512,1462>,
<1462,1463,1513>,
<1514,1513,1463>,
<1463,1464,1514>,
<1515,1514,1464>,
<1464,1465,1515>,
<1516,1515,1465>,
<1465,1466,1516>,
<1517,1516,1466>,
<1466,1467,1517>,
<1518,1517,1467>,
<1467,1468,1518>,
<1519,1518,1468>,
<1468,1469,1519>,
<1520,1519,1469>,
<1469,1470,1520>,
<1521,1520,1470>,
<1470,1471,1521>,
<1522,1521,1471>,
<1471,1472,1522>,
<1523,1522,1472>,
<1472,1473,1523>,
<1524,1523,1473>,
<1473,1474,1524>,
<1525,1524,1474>,
<1474,1475,1525>,
<1526,1525,1475>,
<1475,1476,1526>,
<1527,1526,1476>,
<1476,1477,1527>,
<1528,1527,1477>,
<1477,1478,1528>,
<1529,1528,1478>,
<1479,1480,1530>,
<1531,1530,1480>,
<1480,1481,1531>,
<1532,1531,1481>,
<1481,1482,1532>,
<1533,1532,1482>,
<1482,1483,1533>,
<1534,1533,1483>,
<1483,1484,1534>,
<1535,1534,1484>,
<1484,1485,1535>,
<1536,1535,1485>,
<1485,1486,1536>,
<1537,1536,1486>,
<1486,1487,1537>,
<1538,1537,1487>,
<1487,1488,1538>,
<1539,1538,1488>,

<1488,1489,1539>,
<1540,1539,1489>,
<1489,1490,1540>,
<1541,1540,1490>,
<1490,1491,1541>,
<1542,1541,1491>,
<1491,1492,1542>,
<1543,1542,1492>,
<1492,1493,1543>,
<1544,1543,1493>,
<1493,1494,1544>,
<1545,1544,1494>,
<1494,1495,1545>,
<1546,1545,1495>,
<1495,1496,1546>,
<1547,1546,1496>,
<1496,1497,1547>,
<1548,1547,1497>,
<1497,1498,1548>,
<1549,1548,1498>,
<1498,1499,1549>,
<1550,1549,1499>,
<1499,1500,1550>,
<1551,1550,1500>,
<1500,1501,1551>,
<1552,1551,1501>,
<1501,1502,1552>,
<1553,1552,1502>,
<1502,1503,1553>,
<1554,1553,1503>,
<1503,1504,1554>,
<1555,1554,1504>,
<1504,1505,1555>,
<1556,1555,1505>,
<1505,1506,1556>,
<1557,1556,1506>,
<1506,1507,1557>,
<1558,1557,1507>,
<1507,1508,1558>,
<1559,1558,1508>,
<1508,1509,1559>,
<1560,1559,1509>,
<1509,1510,1560>,
<1561,1560,1510>,
<1510,1511,1561>,
<1562,1561,1511>,
<1511,1512,1562>,
<1563,1562,1512>,
<1512,1513,1563>,
<1564,1563,1513>,
<1513,1514,1564>,
<1565,1564,1514>,
<1514,1515,1565>,
<1566,1565,1515>,
<1515,1516,1566>,
<1567,1566,1516>,

<1516,1517,1567>,
<1568,1567,1517>,
<1517,1518,1568>,
<1569,1568,1518>,
<1518,1519,1569>,
<1570,1569,1519>,
<1519,1520,1570>,
<1571,1570,1520>,
<1520,1521,1571>,
<1572,1571,1521>,
<1521,1522,1572>,
<1573,1572,1522>,
<1522,1523,1573>,
<1574,1573,1523>,
<1523,1524,1574>,
<1575,1574,1524>,
<1524,1525,1575>,
<1576,1575,1525>,
<1525,1526,1576>,
<1577,1576,1526>,
<1526,1527,1577>,
<1578,1577,1527>,
<1527,1528,1578>,
<1579,1578,1528>,
<1528,1529,1579>,
<1580,1579,1529>,
<1530,1531,1581>,
<1582,1581,1531>,
<1531,1532,1582>,
<1583,1582,1532>,
<1532,1533,1583>,
<1584,1583,1533>,
<1533,1534,1584>,
<1585,1584,1534>,
<1534,1535,1585>,
<1586,1585,1535>,
<1535,1536,1586>,
<1587,1586,1536>,
<1536,1537,1587>,
<1588,1587,1537>,
<1537,1538,1588>,
<1589,1588,1538>,
<1538,1539,1589>,
<1590,1589,1539>,
<1539,1540,1590>,
<1591,1590,1540>,
<1540,1541,1591>,
<1592,1591,1541>,
<1541,1542,1592>,
<1593,1592,1542>,
<1542,1543,1593>,
<1594,1593,1543>,
<1543,1544,1594>,
<1595,1594,1544>,
<1544,1545,1595>,
<1596,1595,1545>,

<1545,1546,1596>,
<1597,1596,1546>,
<1546,1547,1597>,
<1598,1597,1547>,
<1547,1548,1598>,
<1599,1598,1548>,
<1548,1549,1599>,
<1600,1599,1549>,
<1549,1550,1600>,
<1601,1600,1550>,
<1550,1551,1601>,
<1602,1601,1551>,
<1551,1552,1602>,
<1603,1602,1552>,
<1552,1553,1603>,
<1604,1603,1553>,
<1553,1554,1604>,
<1605,1604,1554>,
<1554,1555,1605>,
<1606,1605,1555>,
<1555,1556,1606>,
<1607,1606,1556>,
<1556,1557,1607>,
<1608,1607,1557>,
<1557,1558,1608>,
<1609,1608,1558>,
<1558,1559,1609>,
<1610,1609,1559>,
<1559,1560,1610>,
<1611,1610,1560>,
<1560,1561,1611>,
<1612,1611,1561>,
<1561,1562,1612>,
<1613,1612,1562>,
<1562,1563,1613>,
<1614,1613,1563>,
<1563,1564,1614>,
<1615,1614,1564>,
<1564,1565,1615>,
<1616,1615,1565>,
<1565,1566,1616>,
<1617,1616,1566>,
<1566,1567,1617>,
<1618,1617,1567>,
<1567,1568,1618>,
<1619,1618,1568>,
<1568,1569,1619>,
<1620,1619,1569>,
<1569,1570,1620>,
<1621,1620,1570>,
<1570,1571,1621>,
<1622,1621,1571>,
<1571,1572,1622>,
<1623,1622,1572>,
<1572,1573,1623>,
<1624,1623,1573>,

<1573,1574,1624>,
<1625,1624,1574>,
<1574,1575,1625>,
<1626,1625,1575>,
<1575,1576,1626>,
<1627,1626,1576>,
<1576,1577,1627>,
<1628,1627,1577>,
<1577,1578,1628>,
<1629,1628,1578>,
<1578,1579,1629>,
<1630,1629,1579>,
<1579,1580,1630>,
<1631,1630,1580>,
<1581,1582,1632>,
<1633,1632,1582>,
<1582,1583,1633>,
<1634,1633,1583>,
<1583,1584,1634>,
<1635,1634,1584>,
<1584,1585,1635>,
<1636,1635,1585>,
<1585,1586,1636>,
<1637,1636,1586>,
<1586,1587,1637>,
<1638,1637,1587>,
<1587,1588,1638>,
<1639,1638,1588>,
<1588,1589,1639>,
<1640,1639,1589>,
<1589,1590,1640>,
<1641,1640,1590>,
<1590,1591,1641>,
<1642,1641,1591>,
<1591,1592,1642>,
<1643,1642,1592>,
<1592,1593,1643>,
<1644,1643,1593>,
<1593,1594,1644>,
<1645,1644,1594>,
<1594,1595,1645>,
<1646,1645,1595>,
<1595,1596,1646>,
<1647,1646,1596>,
<1596,1597,1647>,
<1648,1647,1597>,
<1597,1598,1648>,
<1649,1648,1598>,
<1598,1599,1649>,
<1650,1649,1599>,
<1599,1600,1650>,
<1651,1650,1600>,
<1600,1601,1651>,
<1652,1651,1601>,
<1601,1602,1652>,
<1653,1652,1602>,

<1602,1603,1653>,
<1654,1653,1603>,
<1603,1604,1654>,
<1655,1654,1604>,
<1604,1605,1655>,
<1656,1655,1605>,
<1605,1606,1656>,
<1657,1656,1606>,
<1606,1607,1657>,
<1658,1657,1607>,
<1607,1608,1658>,
<1659,1658,1608>,
<1608,1609,1659>,
<1660,1659,1609>,
<1609,1610,1660>,
<1661,1660,1610>,
<1610,1611,1661>,
<1662,1661,1611>,
<1611,1612,1662>,
<1663,1662,1612>,
<1612,1613,1663>,
<1664,1663,1613>,
<1613,1614,1664>,
<1665,1664,1614>,
<1614,1615,1665>,
<1666,1665,1615>,
<1615,1616,1666>,
<1667,1666,1616>,
<1616,1617,1667>,
<1668,1667,1617>,
<1617,1618,1668>,
<1669,1668,1618>,
<1618,1619,1669>,
<1670,1669,1619>,
<1619,1620,1670>,
<1671,1670,1620>,
<1620,1621,1671>,
<1672,1671,1621>,
<1621,1622,1672>,
<1673,1672,1622>,
<1622,1623,1673>,
<1674,1673,1623>,
<1623,1624,1674>,
<1675,1674,1624>,
<1624,1625,1675>,
<1676,1675,1625>,
<1625,1626,1676>,
<1677,1676,1626>,
<1626,1627,1677>,
<1678,1677,1627>,
<1627,1628,1678>,
<1679,1678,1628>,
<1628,1629,1679>,
<1680,1679,1629>,
<1629,1630,1680>,
<1681,1680,1630>,

<1630,1631,1681>,
<1682,1681,1631>,
<1632,1633,1683>,
<1684,1683,1633>,
<1633,1634,1684>,
<1685,1684,1634>,
<1634,1635,1685>,
<1686,1685,1635>,
<1635,1636,1686>,
<1687,1686,1636>,
<1636,1637,1687>,
<1688,1687,1637>,
<1637,1638,1688>,
<1689,1688,1638>,
<1638,1639,1689>,
<1690,1689,1639>,
<1639,1640,1690>,
<1691,1690,1640>,
<1640,1641,1691>,
<1692,1691,1641>,
<1641,1642,1692>,
<1693,1692,1642>,
<1642,1643,1693>,
<1694,1693,1643>,
<1643,1644,1694>,
<1695,1694,1644>,
<1644,1645,1695>,
<1696,1695,1645>,
<1645,1646,1696>,
<1697,1696,1646>,
<1646,1647,1697>,
<1698,1697,1647>,
<1647,1648,1698>,
<1699,1698,1648>,
<1648,1649,1699>,
<1700,1699,1649>,
<1649,1650,1700>,
<1701,1700,1650>,
<1650,1651,1701>,
<1702,1701,1651>,
<1651,1652,1702>,
<1703,1702,1652>,
<1652,1653,1703>,
<1704,1703,1653>,
<1653,1654,1704>,
<1705,1704,1654>,
<1654,1655,1705>,
<1706,1705,1655>,
<1655,1656,1706>,
<1707,1706,1656>,
<1656,1657,1707>,
<1708,1707,1657>,
<1657,1658,1708>,
<1709,1708,1658>,
<1658,1659,1709>,
<1710,1709,1659>,

<1659,1660,1710>,
<1711,1710,1660>,
<1660,1661,1711>,
<1712,1711,1661>,
<1661,1662,1712>,
<1713,1712,1662>,
<1662,1663,1713>,
<1714,1713,1663>,
<1663,1664,1714>,
<1715,1714,1664>,
<1664,1665,1715>,
<1716,1715,1665>,
<1665,1666,1716>,
<1717,1716,1666>,
<1666,1667,1717>,
<1718,1717,1667>,
<1667,1668,1718>,
<1719,1718,1668>,
<1668,1669,1719>,
<1720,1719,1669>,
<1669,1670,1720>,
<1721,1720,1670>,
<1670,1671,1721>,
<1722,1721,1671>,
<1671,1672,1722>,
<1723,1722,1672>,
<1672,1673,1723>,
<1724,1723,1673>,
<1673,1674,1724>,
<1725,1724,1674>,
<1674,1675,1725>,
<1726,1725,1675>,
<1675,1676,1726>,
<1727,1726,1676>,
<1676,1677,1727>,
<1728,1727,1677>,
<1677,1678,1728>,
<1729,1728,1678>,
<1678,1679,1729>,
<1730,1729,1679>,
<1679,1680,1730>,
<1731,1730,1680>,
<1680,1681,1731>,
<1732,1731,1681>,
<1681,1682,1732>,
<1733,1732,1682>,
<1683,1684,1734>,
<1735,1734,1684>,
<1684,1685,1735>,
<1736,1735,1685>,
<1685,1686,1736>,
<1737,1736,1686>,
<1686,1687,1737>,
<1738,1737,1687>,
<1687,1688,1738>,
<1739,1738,1688>,

<1688,1689,1739>,
<1740,1739,1689>,
<1689,1690,1740>,
<1741,1740,1690>,
<1690,1691,1741>,
<1742,1741,1691>,
<1691,1692,1742>,
<1743,1742,1692>,
<1692,1693,1743>,
<1744,1743,1693>,
<1693,1694,1744>,
<1745,1744,1694>,
<1694,1695,1745>,
<1746,1745,1695>,
<1695,1696,1746>,
<1747,1746,1696>,
<1696,1697,1747>,
<1748,1747,1697>,
<1697,1698,1748>,
<1749,1748,1698>,
<1698,1699,1749>,
<1750,1749,1699>,
<1699,1700,1750>,
<1751,1750,1700>,
<1700,1701,1751>,
<1752,1751,1701>,
<1701,1702,1752>,
<1753,1752,1702>,
<1702,1703,1753>,
<1754,1753,1703>,
<1703,1704,1754>,
<1755,1754,1704>,
<1704,1705,1755>,
<1756,1755,1705>,
<1705,1706,1756>,
<1757,1756,1706>,
<1706,1707,1757>,
<1758,1757,1707>,
<1707,1708,1758>,
<1759,1758,1708>,
<1708,1709,1759>,
<1760,1759,1709>,
<1709,1710,1760>,
<1761,1760,1710>,
<1710,1711,1761>,
<1762,1761,1711>,
<1711,1712,1762>,
<1763,1762,1712>,
<1712,1713,1763>,
<1764,1763,1713>,
<1713,1714,1764>,
<1765,1764,1714>,
<1714,1715,1765>,
<1766,1765,1715>,
<1715,1716,1766>,
<1767,1766,1716>,

<1716,1717,1767>,
<1768,1767,1717>,
<1717,1718,1768>,
<1769,1768,1718>,
<1718,1719,1769>,
<1770,1769,1719>,
<1719,1720,1770>,
<1771,1770,1720>,
<1720,1721,1771>,
<1772,1771,1721>,
<1721,1722,1772>,
<1773,1772,1722>,
<1722,1723,1773>,
<1774,1773,1723>,
<1723,1724,1774>,
<1775,1774,1724>,
<1724,1725,1775>,
<1776,1775,1725>,
<1725,1726,1776>,
<1777,1776,1726>,
<1726,1727,1777>,
<1778,1777,1727>,
<1727,1728,1778>,
<1779,1778,1728>,
<1728,1729,1779>,
<1780,1779,1729>,
<1729,1730,1780>,
<1781,1780,1730>,
<1730,1731,1781>,
<1782,1781,1731>,
<1731,1732,1782>,
<1783,1782,1732>,
<1732,1733,1783>,
<1784,1783,1733>,
<1734,1735,1785>,
<1786,1785,1735>,
<1735,1736,1786>,
<1787,1786,1736>,
<1736,1737,1787>,
<1788,1787,1737>,
<1737,1738,1788>,
<1789,1788,1738>,
<1738,1739,1789>,
<1790,1789,1739>,
<1739,1740,1790>,
<1791,1790,1740>,
<1740,1741,1791>,
<1792,1791,1741>,
<1741,1742,1792>,
<1793,1792,1742>,
<1742,1743,1793>,
<1794,1793,1743>,
<1743,1744,1794>,
<1795,1794,1744>,
<1744,1745,1795>,
<1796,1795,1745>,

<1745,1746,1796>,
<1797,1796,1746>,
<1746,1747,1797>,
<1798,1797,1747>,
<1747,1748,1798>,
<1799,1798,1748>,
<1748,1749,1799>,
<1800,1799,1749>,
<1749,1750,1800>,
<1801,1800,1750>,
<1750,1751,1801>,
<1802,1801,1751>,
<1751,1752,1802>,
<1803,1802,1752>,
<1752,1753,1803>,
<1804,1803,1753>,
<1753,1754,1804>,
<1805,1804,1754>,
<1754,1755,1805>,
<1806,1805,1755>,
<1755,1756,1806>,
<1807,1806,1756>,
<1756,1757,1807>,
<1808,1807,1757>,
<1757,1758,1808>,
<1809,1808,1758>,
<1758,1759,1809>,
<1810,1809,1759>,
<1759,1760,1810>,
<1811,1810,1760>,
<1760,1761,1811>,
<1812,1811,1761>,
<1761,1762,1812>,
<1813,1812,1762>,
<1762,1763,1813>,
<1814,1813,1763>,
<1763,1764,1814>,
<1815,1814,1764>,
<1764,1765,1815>,
<1816,1815,1765>,
<1765,1766,1816>,
<1817,1816,1766>,
<1766,1767,1817>,
<1818,1817,1767>,
<1767,1768,1818>,
<1819,1818,1768>,
<1768,1769,1819>,
<1820,1819,1769>,
<1769,1770,1820>,
<1821,1820,1770>,
<1770,1771,1821>,
<1822,1821,1771>,
<1771,1772,1822>,
<1823,1822,1772>,
<1772,1773,1823>,
<1824,1823,1773>,

<1773,1774,1824>,
<1825,1824,1774>,
<1774,1775,1825>,
<1826,1825,1775>,
<1775,1776,1826>,
<1827,1826,1776>,
<1776,1777,1827>,
<1828,1827,1777>,
<1777,1778,1828>,
<1829,1828,1778>,
<1778,1779,1829>,
<1830,1829,1779>,
<1779,1780,1830>,
<1831,1830,1780>,
<1780,1781,1831>,
<1832,1831,1781>,
<1781,1782,1832>,
<1833,1832,1782>,
<1782,1783,1833>,
<1834,1833,1783>,
<1783,1784,1834>,
<1835,1834,1784>,
<1785,1786,1836>,
<1837,1836,1786>,
<1786,1787,1837>,
<1838,1837,1787>,
<1787,1788,1838>,
<1839,1838,1788>,
<1788,1789,1839>,
<1840,1839,1789>,
<1789,1790,1840>,
<1841,1840,1790>,
<1790,1791,1841>,
<1842,1841,1791>,
<1791,1792,1842>,
<1843,1842,1792>,
<1792,1793,1843>,
<1844,1843,1793>,
<1793,1794,1844>,
<1845,1844,1794>,
<1794,1795,1845>,
<1846,1845,1795>,
<1795,1796,1846>,
<1847,1846,1796>,
<1796,1797,1847>,
<1848,1847,1797>,
<1797,1798,1848>,
<1849,1848,1798>,
<1798,1799,1849>,
<1850,1849,1799>,
<1799,1800,1850>,
<1851,1850,1800>,
<1800,1801,1851>,
<1852,1851,1801>,
<1801,1802,1852>,
<1853,1852,1802>,

<1802,1803,1853>,
<1854,1853,1803>,
<1803,1804,1854>,
<1855,1854,1804>,
<1804,1805,1855>,
<1856,1855,1805>,
<1805,1806,1856>,
<1857,1856,1806>,
<1806,1807,1857>,
<1858,1857,1807>,
<1807,1808,1858>,
<1859,1858,1808>,
<1808,1809,1859>,
<1860,1859,1809>,
<1809,1810,1860>,
<1861,1860,1810>,
<1810,1811,1861>,
<1862,1861,1811>,
<1811,1812,1862>,
<1863,1862,1812>,
<1812,1813,1863>,
<1864,1863,1813>,
<1813,1814,1864>,
<1865,1864,1814>,
<1814,1815,1865>,
<1866,1865,1815>,
<1815,1816,1866>,
<1867,1866,1816>,
<1816,1817,1867>,
<1868,1867,1817>,
<1817,1818,1868>,
<1869,1868,1818>,
<1818,1819,1869>,
<1870,1869,1819>,
<1819,1820,1870>,
<1871,1870,1820>,
<1820,1821,1871>,
<1872,1871,1821>,
<1821,1822,1872>,
<1873,1872,1822>,
<1822,1823,1873>,
<1874,1873,1823>,
<1823,1824,1874>,
<1875,1874,1824>,
<1824,1825,1875>,
<1876,1875,1825>,
<1825,1826,1876>,
<1877,1876,1826>,
<1826,1827,1877>,
<1878,1877,1827>,
<1827,1828,1878>,
<1879,1878,1828>,
<1828,1829,1879>,
<1880,1879,1829>,
<1829,1830,1880>,
<1881,1880,1830>,

<1830,1831,1881>,
<1882,1881,1831>,
<1831,1832,1882>,
<1883,1882,1832>,
<1832,1833,1883>,
<1884,1883,1833>,
<1833,1834,1884>,
<1885,1884,1834>,
<1834,1835,1885>,
<1886,1885,1835>,
<1836,1837,1887>,
<1888,1887,1837>,
<1837,1838,1888>,
<1889,1888,1838>,
<1838,1839,1889>,
<1890,1889,1839>,
<1839,1840,1890>,
<1891,1890,1840>,
<1840,1841,1891>,
<1892,1891,1841>,
<1841,1842,1892>,
<1893,1892,1842>,
<1842,1843,1893>,
<1894,1893,1843>,
<1843,1844,1894>,
<1895,1894,1844>,
<1844,1845,1895>,
<1896,1895,1845>,
<1845,1846,1896>,
<1897,1896,1846>,
<1846,1847,1897>,
<1898,1897,1847>,
<1847,1848,1898>,
<1899,1898,1848>,
<1848,1849,1899>,
<1900,1899,1849>,
<1849,1850,1900>,
<1901,1900,1850>,
<1850,1851,1901>,
<1902,1901,1851>,
<1851,1852,1902>,
<1903,1902,1852>,
<1852,1853,1903>,
<1904,1903,1853>,
<1853,1854,1904>,
<1905,1904,1854>,
<1854,1855,1905>,
<1906,1905,1855>,
<1855,1856,1906>,
<1907,1906,1856>,
<1856,1857,1907>,
<1908,1907,1857>,
<1857,1858,1908>,
<1909,1908,1858>,
<1858,1859,1909>,
<1910,1909,1859>,

<1859,1860,1910>,
<1911,1910,1860>,
<1860,1861,1911>,
<1912,1911,1861>,
<1861,1862,1912>,
<1913,1912,1862>,
<1862,1863,1913>,
<1914,1913,1863>,
<1863,1864,1914>,
<1915,1914,1864>,
<1864,1865,1915>,
<1916,1915,1865>,
<1865,1866,1916>,
<1917,1916,1866>,
<1866,1867,1917>,
<1918,1917,1867>,
<1867,1868,1918>,
<1919,1918,1868>,
<1868,1869,1919>,
<1920,1919,1869>,
<1869,1870,1920>,
<1921,1920,1870>,
<1870,1871,1921>,
<1922,1921,1871>,
<1871,1872,1922>,
<1923,1922,1872>,
<1872,1873,1923>,
<1924,1923,1873>,
<1873,1874,1924>,
<1925,1924,1874>,
<1874,1875,1925>,
<1926,1925,1875>,
<1875,1876,1926>,
<1927,1926,1876>,
<1876,1877,1927>,
<1928,1927,1877>,
<1877,1878,1928>,
<1929,1928,1878>,
<1878,1879,1929>,
<1930,1929,1879>,
<1879,1880,1930>,
<1931,1930,1880>,
<1880,1881,1931>,
<1932,1931,1881>,
<1881,1882,1932>,
<1933,1932,1882>,
<1882,1883,1933>,
<1934,1933,1883>,
<1883,1884,1934>,
<1935,1934,1884>,
<1884,1885,1935>,
<1936,1935,1885>,
<1885,1886,1936>,
<1937,1936,1886>,
<1887,1888,1938>,
<1939,1938,1888>,

<1888,1889,1939>,
<1940,1939,1889>,
<1889,1890,1940>,
<1941,1940,1890>,
<1890,1891,1941>,
<1942,1941,1891>,
<1891,1892,1942>,
<1943,1942,1892>,
<1892,1893,1943>,
<1944,1943,1893>,
<1893,1894,1944>,
<1945,1944,1894>,
<1894,1895,1945>,
<1946,1945,1895>,
<1895,1896,1946>,
<1947,1946,1896>,
<1896,1897,1947>,
<1948,1947,1897>,
<1897,1898,1948>,
<1949,1948,1898>,
<1898,1899,1949>,
<1950,1949,1899>,
<1899,1900,1950>,
<1951,1950,1900>,
<1900,1901,1951>,
<1952,1951,1901>,
<1901,1902,1952>,
<1953,1952,1902>,
<1902,1903,1953>,
<1954,1953,1903>,
<1903,1904,1954>,
<1955,1954,1904>,
<1904,1905,1955>,
<1956,1955,1905>,
<1905,1906,1956>,
<1957,1956,1906>,
<1906,1907,1957>,
<1958,1957,1907>,
<1907,1908,1958>,
<1959,1958,1908>,
<1908,1909,1959>,
<1960,1959,1909>,
<1909,1910,1960>,
<1961,1960,1910>,
<1910,1911,1961>,
<1962,1961,1911>,
<1911,1912,1962>,
<1963,1962,1912>,
<1912,1913,1963>,
<1964,1963,1913>,
<1913,1914,1964>,
<1965,1964,1914>,
<1914,1915,1965>,
<1966,1965,1915>,
<1915,1916,1966>,
<1967,1966,1916>,

<1916,1917,1967>,
<1968,1967,1917>,
<1917,1918,1968>,
<1969,1968,1918>,
<1918,1919,1969>,
<1970,1969,1919>,
<1919,1920,1970>,
<1971,1970,1920>,
<1920,1921,1971>,
<1972,1971,1921>,
<1921,1922,1972>,
<1973,1972,1922>,
<1922,1923,1973>,
<1974,1973,1923>,
<1923,1924,1974>,
<1975,1974,1924>,
<1924,1925,1975>,
<1976,1975,1925>,
<1925,1926,1976>,
<1977,1976,1926>,
<1926,1927,1977>,
<1978,1977,1927>,
<1927,1928,1978>,
<1979,1978,1928>,
<1928,1929,1979>,
<1980,1979,1929>,
<1929,1930,1980>,
<1981,1980,1930>,
<1930,1931,1981>,
<1982,1981,1931>,
<1931,1932,1982>,
<1983,1982,1932>,
<1932,1933,1983>,
<1984,1983,1933>,
<1933,1934,1984>,
<1985,1984,1934>,
<1934,1935,1985>,
<1986,1985,1935>,
<1935,1936,1986>,
<1987,1986,1936>,
<1936,1937,1987>,
<1988,1987,1937>,
<1938,1939,1989>,
<1990,1989,1939>,
<1939,1940,1990>,
<1991,1990,1940>,
<1940,1941,1991>,
<1992,1991,1941>,
<1941,1942,1992>,
<1993,1992,1942>,
<1942,1943,1993>,
<1994,1993,1943>,
<1943,1944,1994>,
<1995,1994,1944>,
<1944,1945,1995>,
<1996,1995,1945>,

<1945, 1946, 1996>,
<1997, 1996, 1946>,
<1946, 1947, 1997>,
<1998, 1997, 1947>,
<1947, 1948, 1998>,
<1999, 1998, 1948>,
<1948, 1949, 1999>,
<2000, 1999, 1949>,
<1949, 1950, 2000>,
<2001, 2000, 1950>,
<1950, 1951, 2001>,
<2002, 2001, 1951>,
<1951, 1952, 2002>,
<2003, 2002, 1952>,
<1952, 1953, 2003>,
<2004, 2003, 1953>,
<1953, 1954, 2004>,
<2005, 2004, 1954>,
<1954, 1955, 2005>,
<2006, 2005, 1955>,
<1955, 1956, 2006>,
<2007, 2006, 1956>,
<1956, 1957, 2007>,
<2008, 2007, 1957>,
<1957, 1958, 2008>,
<2009, 2008, 1958>,
<1958, 1959, 2009>,
<2010, 2009, 1959>,
<1959, 1960, 2010>,
<2011, 2010, 1960>,
<1960, 1961, 2011>,
<2012, 2011, 1961>,
<1961, 1962, 2012>,
<2013, 2012, 1962>,
<1962, 1963, 2013>,
<2014, 2013, 1963>,
<1963, 1964, 2014>,
<2015, 2014, 1964>,
<1964, 1965, 2015>,
<2016, 2015, 1965>,
<1965, 1966, 2016>,
<2017, 2016, 1966>,
<1966, 1967, 2017>,
<2018, 2017, 1967>,
<1967, 1968, 2018>,
<2019, 2018, 1968>,
<1968, 1969, 2019>,
<2020, 2019, 1969>,
<1969, 1970, 2020>,
<2021, 2020, 1970>,
<1970, 1971, 2021>,
<2022, 2021, 1971>,
<1971, 1972, 2022>,
<2023, 2022, 1972>,
<1972, 1973, 2023>,
<2024, 2023, 1973>,

<1973,1974,2024>,
<2025,2024,1974>,
<1974,1975,2025>,
<2026,2025,1975>,
<1975,1976,2026>,
<2027,2026,1976>,
<1976,1977,2027>,
<2028,2027,1977>,
<1977,1978,2028>,
<2029,2028,1978>,
<1978,1979,2029>,
<2030,2029,1979>,
<1979,1980,2030>,
<2031,2030,1980>,
<1980,1981,2031>,
<2032,2031,1981>,
<1981,1982,2032>,
<2033,2032,1982>,
<1982,1983,2033>,
<2034,2033,1983>,
<1983,1984,2034>,
<2035,2034,1984>,
<1984,1985,2035>,
<2036,2035,1985>,
<1985,1986,2036>,
<2037,2036,1986>,
<1986,1987,2037>,
<2038,2037,1987>,
<1987,1988,2038>,
<2039,2038,1988>,
<1989,1990,2040>,
<2041,2040,1990>,
<1990,1991,2041>,
<2042,2041,1991>,
<1991,1992,2042>,
<2043,2042,1992>,
<1992,1993,2043>,
<2044,2043,1993>,
<1993,1994,2044>,
<2045,2044,1994>,
<1994,1995,2045>,
<2046,2045,1995>,
<1995,1996,2046>,
<2047,2046,1996>,
<1996,1997,2047>,
<2048,2047,1997>,
<1997,1998,2048>,
<2049,2048,1998>,
<1998,1999,2049>,
<2050,2049,1999>,
<1999,2000,2050>,
<2051,2050,2000>,
<2000,2001,2051>,
<2052,2051,2001>,
<2001,2002,2052>,
<2053,2052,2002>,

<2002,2003,2053>,
<2054,2053,2003>,
<2003,2004,2054>,
<2055,2054,2004>,
<2004,2005,2055>,
<2056,2055,2005>,
<2005,2006,2056>,
<2057,2056,2006>,
<2006,2007,2057>,
<2058,2057,2007>,
<2007,2008,2058>,
<2059,2058,2008>,
<2008,2009,2059>,
<2060,2059,2009>,
<2009,2010,2060>,
<2061,2060,2010>,
<2010,2011,2061>,
<2062,2061,2011>,
<2011,2012,2062>,
<2063,2062,2012>,
<2012,2013,2063>,
<2064,2063,2013>,
<2013,2014,2064>,
<2065,2064,2014>,
<2014,2015,2065>,
<2066,2065,2015>,
<2015,2016,2066>,
<2067,2066,2016>,
<2016,2017,2067>,
<2068,2067,2017>,
<2017,2018,2068>,
<2069,2068,2018>,
<2018,2019,2069>,
<2070,2069,2019>,
<2019,2020,2070>,
<2071,2070,2020>,
<2020,2021,2071>,
<2072,2071,2021>,
<2021,2022,2072>,
<2073,2072,2022>,
<2022,2023,2073>,
<2074,2073,2023>,
<2023,2024,2074>,
<2075,2074,2024>,
<2024,2025,2075>,
<2076,2075,2025>,
<2025,2026,2076>,
<2077,2076,2026>,
<2026,2027,2077>,
<2078,2077,2027>,
<2027,2028,2078>,
<2079,2078,2028>,
<2028,2029,2079>,
<2080,2079,2029>,
<2029,2030,2080>,
<2081,2080,2030>,

<2030,2031,2081>,
<2082,2081,2031>,
<2031,2032,2082>,
<2083,2082,2032>,
<2032,2033,2083>,
<2084,2083,2033>,
<2033,2034,2084>,
<2085,2084,2034>,
<2034,2035,2085>,
<2086,2085,2035>,
<2035,2036,2086>,
<2087,2086,2036>,
<2036,2037,2087>,
<2088,2087,2037>,
<2037,2038,2088>,
<2089,2088,2038>,
<2038,2039,2089>,
<2090,2089,2039>,
<2040,2041,2091>,
<2092,2091,2041>,
<2041,2042,2092>,
<2093,2092,2042>,
<2042,2043,2093>,
<2094,2093,2043>,
<2043,2044,2094>,
<2095,2094,2044>,
<2044,2045,2095>,
<2096,2095,2045>,
<2045,2046,2096>,
<2097,2096,2046>,
<2046,2047,2097>,
<2098,2097,2047>,
<2047,2048,2098>,
<2099,2098,2048>,
<2048,2049,2099>,
<2100,2099,2049>,
<2049,2050,2100>,
<2101,2100,2050>,
<2050,2051,2101>,
<2102,2101,2051>,
<2051,2052,2102>,
<2103,2102,2052>,
<2052,2053,2103>,
<2104,2103,2053>,
<2053,2054,2104>,
<2105,2104,2054>,
<2054,2055,2105>,
<2106,2105,2055>,
<2055,2056,2106>,
<2107,2106,2056>,
<2056,2057,2107>,
<2108,2107,2057>,
<2057,2058,2108>,
<2109,2108,2058>,
<2058,2059,2109>,
<2110,2109,2059>,

<2059,2060,2110>,
<2111,2110,2060>,
<2060,2061,2111>,
<2112,2111,2061>,
<2061,2062,2112>,
<2113,2112,2062>,
<2062,2063,2113>,
<2114,2113,2063>,
<2063,2064,2114>,
<2115,2114,2064>,
<2064,2065,2115>,
<2116,2115,2065>,
<2065,2066,2116>,
<2117,2116,2066>,
<2066,2067,2117>,
<2118,2117,2067>,
<2067,2068,2118>,
<2119,2118,2068>,
<2068,2069,2119>,
<2120,2119,2069>,
<2069,2070,2120>,
<2121,2120,2070>,
<2070,2071,2121>,
<2122,2121,2071>,
<2071,2072,2122>,
<2123,2122,2072>,
<2072,2073,2123>,
<2124,2123,2073>,
<2073,2074,2124>,
<2125,2124,2074>,
<2074,2075,2125>,
<2126,2125,2075>,
<2075,2076,2126>,
<2127,2126,2076>,
<2076,2077,2127>,
<2128,2127,2077>,
<2077,2078,2128>,
<2129,2128,2078>,
<2078,2079,2129>,
<2130,2129,2079>,
<2079,2080,2130>,
<2131,2130,2080>,
<2080,2081,2131>,
<2132,2131,2081>,
<2081,2082,2132>,
<2133,2132,2082>,
<2082,2083,2133>,
<2134,2133,2083>,
<2083,2084,2134>,
<2135,2134,2084>,
<2084,2085,2135>,
<2136,2135,2085>,
<2085,2086,2136>,
<2137,2136,2086>,
<2086,2087,2137>,
<2138,2137,2087>,

<2087,2088,2138>,
<2139,2138,2088>,
<2088,2089,2139>,
<2140,2139,2089>,
<2089,2090,2140>,
<2141,2140,2090>,
<2091,2092,2142>,
<2143,2142,2092>,
<2092,2093,2143>,
<2144,2143,2093>,
<2093,2094,2144>,
<2145,2144,2094>,
<2094,2095,2145>,
<2146,2145,2095>,
<2095,2096,2146>,
<2147,2146,2096>,
<2096,2097,2147>,
<2148,2147,2097>,
<2097,2098,2148>,
<2149,2148,2098>,
<2098,2099,2149>,
<2150,2149,2099>,
<2099,2100,2150>,
<2151,2150,2100>,
<2100,2101,2151>,
<2152,2151,2101>,
<2101,2102,2152>,
<2153,2152,2102>,
<2102,2103,2153>,
<2154,2153,2103>,
<2103,2104,2154>,
<2155,2154,2104>,
<2104,2105,2155>,
<2156,2155,2105>,
<2105,2106,2156>,
<2157,2156,2106>,
<2106,2107,2157>,
<2158,2157,2107>,
<2107,2108,2158>,
<2159,2158,2108>,
<2108,2109,2159>,
<2160,2159,2109>,
<2109,2110,2160>,
<2161,2160,2110>,
<2110,2111,2161>,
<2162,2161,2111>,
<2111,2112,2162>,
<2163,2162,2112>,
<2112,2113,2163>,
<2164,2163,2113>,
<2113,2114,2164>,
<2165,2164,2114>,
<2114,2115,2165>,
<2166,2165,2115>,
<2115,2116,2166>,
<2167,2166,2116>,

<2116,2117,2167>,
<2168,2167,2117>,
<2117,2118,2168>,
<2169,2168,2118>,
<2118,2119,2169>,
<2170,2169,2119>,
<2119,2120,2170>,
<2171,2170,2120>,
<2120,2121,2171>,
<2172,2171,2121>,
<2121,2122,2172>,
<2173,2172,2122>,
<2122,2123,2173>,
<2174,2173,2123>,
<2123,2124,2174>,
<2175,2174,2124>,
<2124,2125,2175>,
<2176,2175,2125>,
<2125,2126,2176>,
<2177,2176,2126>,
<2126,2127,2177>,
<2178,2177,2127>,
<2127,2128,2178>,
<2179,2178,2128>,
<2128,2129,2179>,
<2180,2179,2129>,
<2129,2130,2180>,
<2181,2180,2130>,
<2130,2131,2181>,
<2182,2181,2131>,
<2131,2132,2182>,
<2183,2182,2132>,
<2132,2133,2183>,
<2184,2183,2133>,
<2133,2134,2184>,
<2185,2184,2134>,
<2134,2135,2185>,
<2186,2185,2135>,
<2135,2136,2186>,
<2187,2186,2136>,
<2136,2137,2187>,
<2188,2187,2137>,
<2137,2138,2188>,
<2189,2188,2138>,
<2138,2139,2189>,
<2190,2189,2139>,
<2139,2140,2190>,
<2191,2190,2140>,
<2140,2141,2191>,
<2192,2191,2141>,
<2142,2143,2193>,
<2194,2193,2143>,
<2143,2144,2194>,
<2195,2194,2144>,
<2144,2145,2195>,
<2196,2195,2145>,

<2145,2146,2196>,
<2197,2196,2146>,
<2146,2147,2197>,
<2198,2197,2147>,
<2147,2148,2198>,
<2199,2198,2148>,
<2148,2149,2199>,
<2200,2199,2149>,
<2149,2150,2200>,
<2201,2200,2150>,
<2150,2151,2201>,
<2202,2201,2151>,
<2151,2152,2202>,
<2203,2202,2152>,
<2152,2153,2203>,
<2204,2203,2153>,
<2153,2154,2204>,
<2205,2204,2154>,
<2154,2155,2205>,
<2206,2205,2155>,
<2155,2156,2206>,
<2207,2206,2156>,
<2156,2157,2207>,
<2208,2207,2157>,
<2157,2158,2208>,
<2209,2208,2158>,
<2158,2159,2209>,
<2210,2209,2159>,
<2159,2160,2210>,
<2211,2210,2160>,
<2160,2161,2211>,
<2212,2211,2161>,
<2161,2162,2212>,
<2213,2212,2162>,
<2162,2163,2213>,
<2214,2213,2163>,
<2163,2164,2214>,
<2215,2214,2164>,
<2164,2165,2215>,
<2216,2215,2165>,
<2165,2166,2216>,
<2217,2216,2166>,
<2166,2167,2217>,
<2218,2217,2167>,
<2167,2168,2218>,
<2219,2218,2168>,
<2168,2169,2219>,
<2220,2219,2169>,
<2169,2170,2220>,
<2221,2220,2170>,
<2170,2171,2221>,
<2222,2221,2171>,
<2171,2172,2222>,
<2223,2222,2172>,
<2172,2173,2223>,
<2224,2223,2173>,

<2173,2174,2224>,
<2225,2224,2174>,
<2174,2175,2225>,
<2226,2225,2175>,
<2175,2176,2226>,
<2227,2226,2176>,
<2176,2177,2227>,
<2228,2227,2177>,
<2177,2178,2228>,
<2229,2228,2178>,
<2178,2179,2229>,
<2230,2229,2179>,
<2179,2180,2230>,
<2231,2230,2180>,
<2180,2181,2231>,
<2232,2231,2181>,
<2181,2182,2232>,
<2233,2232,2182>,
<2182,2183,2233>,
<2234,2233,2183>,
<2183,2184,2234>,
<2235,2234,2184>,
<2184,2185,2235>,
<2236,2235,2185>,
<2185,2186,2236>,
<2237,2236,2186>,
<2186,2187,2237>,
<2238,2237,2187>,
<2187,2188,2238>,
<2239,2238,2188>,
<2188,2189,2239>,
<2240,2239,2189>,
<2189,2190,2240>,
<2241,2240,2190>,
<2190,2191,2241>,
<2242,2241,2191>,
<2191,2192,2242>,
<2243,2242,2192>,
<2193,2194,2244>,
<2245,2244,2194>,
<2194,2195,2245>,
<2246,2245,2195>,
<2195,2196,2246>,
<2247,2246,2196>,
<2196,2197,2247>,
<2248,2247,2197>,
<2197,2198,2248>,
<2249,2248,2198>,
<2198,2199,2249>,
<2250,2249,2199>,
<2199,2200,2250>,
<2251,2250,2200>,
<2200,2201,2251>,
<2252,2251,2201>,
<2201,2202,2252>,
<2253,2252,2202>,

<2202,2203,2253>,
<2254,2253,2203>,
<2203,2204,2254>,
<2255,2254,2204>,
<2204,2205,2255>,
<2256,2255,2205>,
<2205,2206,2256>,
<2257,2256,2206>,
<2206,2207,2257>,
<2258,2257,2207>,
<2207,2208,2258>,
<2259,2258,2208>,
<2208,2209,2259>,
<2260,2259,2209>,
<2209,2210,2260>,
<2261,2260,2210>,
<2210,2211,2261>,
<2262,2261,2211>,
<2211,2212,2262>,
<2263,2262,2212>,
<2212,2213,2263>,
<2264,2263,2213>,
<2213,2214,2264>,
<2265,2264,2214>,
<2214,2215,2265>,
<2266,2265,2215>,
<2215,2216,2266>,
<2267,2266,2216>,
<2216,2217,2267>,
<2268,2267,2217>,
<2217,2218,2268>,
<2269,2268,2218>,
<2218,2219,2269>,
<2270,2269,2219>,
<2219,2220,2270>,
<2271,2270,2220>,
<2220,2221,2271>,
<2272,2271,2221>,
<2221,2222,2272>,
<2273,2272,2222>,
<2222,2223,2273>,
<2274,2273,2223>,
<2223,2224,2274>,
<2275,2274,2224>,
<2224,2225,2275>,
<2276,2275,2225>,
<2225,2226,2276>,
<2277,2276,2226>,
<2226,2227,2277>,
<2278,2277,2227>,
<2227,2228,2278>,
<2279,2278,2228>,
<2228,2229,2279>,
<2280,2279,2229>,
<2229,2230,2280>,
<2281,2280,2230>,

<2230,2231,2281>,
<2282,2281,2231>,
<2231,2232,2282>,
<2283,2282,2232>,
<2232,2233,2283>,
<2284,2283,2233>,
<2233,2234,2284>,
<2285,2284,2234>,
<2234,2235,2285>,
<2286,2285,2235>,
<2235,2236,2286>,
<2287,2286,2236>,
<2236,2237,2287>,
<2288,2287,2237>,
<2237,2238,2288>,
<2289,2288,2238>,
<2238,2239,2289>,
<2290,2289,2239>,
<2239,2240,2290>,
<2291,2290,2240>,
<2240,2241,2291>,
<2292,2291,2241>,
<2241,2242,2292>,
<2293,2292,2242>,
<2242,2243,2293>,
<2294,2293,2243>,
<2244,2245,2295>,
<2296,2295,2245>,
<2245,2246,2296>,
<2297,2296,2246>,
<2246,2247,2297>,
<2298,2297,2247>,
<2247,2248,2298>,
<2299,2298,2248>,
<2248,2249,2299>,
<2300,2299,2249>,
<2249,2250,2300>,
<2301,2300,2250>,
<2250,2251,2301>,
<2302,2301,2251>,
<2251,2252,2302>,
<2303,2302,2252>,
<2252,2253,2303>,
<2304,2303,2253>,
<2253,2254,2304>,
<2305,2304,2254>,
<2254,2255,2305>,
<2306,2305,2255>,
<2255,2256,2306>,
<2307,2306,2256>,
<2256,2257,2307>,
<2308,2307,2257>,
<2257,2258,2308>,
<2309,2308,2258>,
<2258,2259,2309>,
<2310,2309,2259>,

<2259,2260,2310>,
<2311,2310,2260>,
<2260,2261,2311>,
<2312,2311,2261>,
<2261,2262,2312>,
<2313,2312,2262>,
<2262,2263,2313>,
<2314,2313,2263>,
<2263,2264,2314>,
<2315,2314,2264>,
<2264,2265,2315>,
<2316,2315,2265>,
<2265,2266,2316>,
<2317,2316,2266>,
<2266,2267,2317>,
<2318,2317,2267>,
<2267,2268,2318>,
<2319,2318,2268>,
<2268,2269,2319>,
<2320,2319,2269>,
<2269,2270,2320>,
<2321,2320,2270>,
<2270,2271,2321>,
<2322,2321,2271>,
<2271,2272,2322>,
<2323,2322,2272>,
<2272,2273,2323>,
<2324,2323,2273>,
<2273,2274,2324>,
<2325,2324,2274>,
<2274,2275,2325>,
<2326,2325,2275>,
<2275,2276,2326>,
<2327,2326,2276>,
<2276,2277,2327>,
<2328,2327,2277>,
<2277,2278,2328>,
<2329,2328,2278>,
<2278,2279,2329>,
<2330,2329,2279>,
<2279,2280,2330>,
<2331,2330,2280>,
<2280,2281,2331>,
<2332,2331,2281>,
<2281,2282,2332>,
<2333,2332,2282>,
<2282,2283,2333>,
<2334,2333,2283>,
<2283,2284,2334>,
<2335,2334,2284>,
<2284,2285,2335>,
<2336,2335,2285>,
<2285,2286,2336>,
<2337,2336,2286>,
<2286,2287,2337>,
<2338,2337,2287>,

<2287,2288,2338>,
<2339,2338,2288>,
<2288,2289,2339>,
<2340,2339,2289>,
<2289,2290,2340>,
<2341,2340,2290>,
<2290,2291,2341>,
<2342,2341,2291>,
<2291,2292,2342>,
<2343,2342,2292>,
<2292,2293,2343>,
<2344,2343,2293>,
<2293,2294,2344>,
<2345,2344,2294>,
<2295,2296,2346>,
<2347,2346,2296>,
<2296,2297,2347>,
<2348,2347,2297>,
<2297,2298,2348>,
<2349,2348,2298>,
<2298,2299,2349>,
<2350,2349,2299>,
<2299,2300,2350>,
<2351,2350,2300>,
<2300,2301,2351>,
<2352,2351,2301>,
<2301,2302,2352>,
<2353,2352,2302>,
<2302,2303,2353>,
<2354,2353,2303>,
<2303,2304,2354>,
<2355,2354,2304>,
<2304,2305,2355>,
<2356,2355,2305>,
<2305,2306,2356>,
<2357,2356,2306>,
<2306,2307,2357>,
<2358,2357,2307>,
<2307,2308,2358>,
<2359,2358,2308>,
<2308,2309,2359>,
<2360,2359,2309>,
<2309,2310,2360>,
<2361,2360,2310>,
<2310,2311,2361>,
<2362,2361,2311>,
<2311,2312,2362>,
<2363,2362,2312>,
<2312,2313,2363>,
<2364,2363,2313>,
<2313,2314,2364>,
<2365,2364,2314>,
<2314,2315,2365>,
<2366,2365,2315>,
<2315,2316,2366>,
<2367,2366,2316>,

<2316,2317,2367>,
<2368,2367,2317>,
<2317,2318,2368>,
<2369,2368,2318>,
<2318,2319,2369>,
<2370,2369,2319>,
<2319,2320,2370>,
<2371,2370,2320>,
<2320,2321,2371>,
<2372,2371,2321>,
<2321,2322,2372>,
<2373,2372,2322>,
<2322,2323,2373>,
<2374,2373,2323>,
<2323,2324,2374>,
<2375,2374,2324>,
<2324,2325,2375>,
<2376,2375,2325>,
<2325,2326,2376>,
<2377,2376,2326>,
<2326,2327,2377>,
<2378,2377,2327>,
<2327,2328,2378>,
<2379,2378,2328>,
<2328,2329,2379>,
<2380,2379,2329>,
<2329,2330,2380>,
<2381,2380,2330>,
<2330,2331,2381>,
<2382,2381,2331>,
<2331,2332,2382>,
<2383,2382,2332>,
<2332,2333,2383>,
<2384,2383,2333>,
<2333,2334,2384>,
<2385,2384,2334>,
<2334,2335,2385>,
<2386,2385,2335>,
<2335,2336,2386>,
<2387,2386,2336>,
<2336,2337,2387>,
<2388,2387,2337>,
<2337,2338,2388>,
<2389,2388,2338>,
<2338,2339,2389>,
<2390,2389,2339>,
<2339,2340,2390>,
<2391,2390,2340>,
<2340,2341,2391>,
<2392,2391,2341>,
<2341,2342,2392>,
<2393,2392,2342>,
<2342,2343,2393>,
<2394,2393,2343>,
<2343,2344,2394>,
<2395,2394,2344>,

<2344,2345,2395>,
<2396,2395,2345>,
<2346,2347,2397>,
<2398,2397,2347>,
<2347,2348,2398>,
<2399,2398,2348>,
<2348,2349,2399>,
<2400,2399,2349>,
<2349,2350,2400>,
<2401,2400,2350>,
<2350,2351,2401>,
<2402,2401,2351>,
<2351,2352,2402>,
<2403,2402,2352>,
<2352,2353,2403>,
<2404,2403,2353>,
<2353,2354,2404>,
<2405,2404,2354>,
<2354,2355,2405>,
<2406,2405,2355>,
<2355,2356,2406>,
<2407,2406,2356>,
<2356,2357,2407>,
<2408,2407,2357>,
<2357,2358,2408>,
<2409,2408,2358>,
<2358,2359,2409>,
<2410,2409,2359>,
<2359,2360,2410>,
<2411,2410,2360>,
<2360,2361,2411>,
<2412,2411,2361>,
<2361,2362,2412>,
<2413,2412,2362>,
<2362,2363,2413>,
<2414,2413,2363>,
<2363,2364,2414>,
<2415,2414,2364>,
<2364,2365,2415>,
<2416,2415,2365>,
<2365,2366,2416>,
<2417,2416,2366>,
<2366,2367,2417>,
<2418,2417,2367>,
<2367,2368,2418>,
<2419,2418,2368>,
<2368,2369,2419>,
<2420,2419,2369>,
<2369,2370,2420>,
<2421,2420,2370>,
<2370,2371,2421>,
<2422,2421,2371>,
<2371,2372,2422>,
<2423,2422,2372>,
<2372,2373,2423>,
<2424,2423,2373>,

<2373,2374,2424>,
<2425,2424,2374>,
<2374,2375,2425>,
<2426,2425,2375>,
<2375,2376,2426>,
<2427,2426,2376>,
<2376,2377,2427>,
<2428,2427,2377>,
<2377,2378,2428>,
<2429,2428,2378>,
<2378,2379,2429>,
<2430,2429,2379>,
<2379,2380,2430>,
<2431,2430,2380>,
<2380,2381,2431>,
<2432,2431,2381>,
<2381,2382,2432>,
<2433,2432,2382>,
<2382,2383,2433>,
<2434,2433,2383>,
<2383,2384,2434>,
<2435,2434,2384>,
<2384,2385,2435>,
<2436,2435,2385>,
<2385,2386,2436>,
<2437,2436,2386>,
<2386,2387,2437>,
<2438,2437,2387>,
<2387,2388,2438>,
<2439,2438,2388>,
<2388,2389,2439>,
<2440,2439,2389>,
<2389,2390,2440>,
<2441,2440,2390>,
<2390,2391,2441>,
<2442,2441,2391>,
<2391,2392,2442>,
<2443,2442,2392>,
<2392,2393,2443>,
<2444,2443,2393>,
<2393,2394,2444>,
<2445,2444,2394>,
<2394,2395,2445>,
<2446,2445,2395>,
<2395,2396,2446>,
<2447,2446,2396>,
<2397,2398,2448>,
<2449,2448,2398>,
<2398,2399,2449>,
<2450,2449,2399>,
<2399,2400,2450>,
<2451,2450,2400>,
<2400,2401,2451>,
<2452,2451,2401>,
<2401,2402,2452>,
<2453,2452,2402>,

<2402,2403,2453>,
<2454,2453,2403>,
<2403,2404,2454>,
<2455,2454,2404>,
<2404,2405,2455>,
<2456,2455,2405>,
<2405,2406,2456>,
<2457,2456,2406>,
<2406,2407,2457>,
<2458,2457,2407>,
<2407,2408,2458>,
<2459,2458,2408>,
<2408,2409,2459>,
<2460,2459,2409>,
<2409,2410,2460>,
<2461,2460,2410>,
<2410,2411,2461>,
<2462,2461,2411>,
<2411,2412,2462>,
<2463,2462,2412>,
<2412,2413,2463>,
<2464,2463,2413>,
<2413,2414,2464>,
<2465,2464,2414>,
<2414,2415,2465>,
<2466,2465,2415>,
<2415,2416,2466>,
<2467,2466,2416>,
<2416,2417,2467>,
<2468,2467,2417>,
<2417,2418,2468>,
<2469,2468,2418>,
<2418,2419,2469>,
<2470,2469,2419>,
<2419,2420,2470>,
<2471,2470,2420>,
<2420,2421,2471>,
<2472,2471,2421>,
<2421,2422,2472>,
<2473,2472,2422>,
<2422,2423,2473>,
<2474,2473,2423>,
<2423,2424,2474>,
<2475,2474,2424>,
<2424,2425,2475>,
<2476,2475,2425>,
<2425,2426,2476>,
<2477,2476,2426>,
<2426,2427,2477>,
<2478,2477,2427>,
<2427,2428,2478>,
<2479,2478,2428>,
<2428,2429,2479>,
<2480,2479,2429>,
<2429,2430,2480>,
<2481,2480,2430>,

<2430,2431,2481>,
<2482,2481,2431>,
<2431,2432,2482>,
<2483,2482,2432>,
<2432,2433,2483>,
<2484,2483,2433>,
<2433,2434,2484>,
<2485,2484,2434>,
<2434,2435,2485>,
<2486,2485,2435>,
<2435,2436,2486>,
<2487,2486,2436>,
<2436,2437,2487>,
<2488,2487,2437>,
<2437,2438,2488>,
<2489,2488,2438>,
<2438,2439,2489>,
<2490,2489,2439>,
<2439,2440,2490>,
<2491,2490,2440>,
<2440,2441,2491>,
<2492,2491,2441>,
<2441,2442,2492>,
<2493,2492,2442>,
<2442,2443,2493>,
<2494,2493,2443>,
<2443,2444,2494>,
<2495,2494,2444>,
<2444,2445,2495>,
<2496,2495,2445>,
<2445,2446,2496>,
<2497,2496,2446>,
<2446,2447,2497>,
<2498,2497,2447>,
<2448,2449,2499>,
<2500,2499,2449>,
<2449,2450,2500>,
<2501,2500,2450>,
<2450,2451,2501>,
<2502,2501,2451>,
<2451,2452,2502>,
<2503,2502,2452>,
<2452,2453,2503>,
<2504,2503,2453>,
<2453,2454,2504>,
<2505,2504,2454>,
<2454,2455,2505>,
<2506,2505,2455>,
<2455,2456,2506>,
<2507,2506,2456>,
<2456,2457,2507>,
<2508,2507,2457>,
<2457,2458,2508>,
<2509,2508,2458>,
<2458,2459,2509>,
<2510,2509,2459>,

<2459,2460,2510>,
<2511,2510,2460>,
<2460,2461,2511>,
<2512,2511,2461>,
<2461,2462,2512>,
<2513,2512,2462>,
<2462,2463,2513>,
<2514,2513,2463>,
<2463,2464,2514>,
<2515,2514,2464>,
<2464,2465,2515>,
<2516,2515,2465>,
<2465,2466,2516>,
<2517,2516,2466>,
<2466,2467,2517>,
<2518,2517,2467>,
<2467,2468,2518>,
<2519,2518,2468>,
<2468,2469,2519>,
<2520,2519,2469>,
<2469,2470,2520>,
<2521,2520,2470>,
<2470,2471,2521>,
<2522,2521,2471>,
<2471,2472,2522>,
<2523,2522,2472>,
<2472,2473,2523>,
<2524,2523,2473>,
<2473,2474,2524>,
<2525,2524,2474>,
<2474,2475,2525>,
<2526,2525,2475>,
<2475,2476,2526>,
<2527,2526,2476>,
<2476,2477,2527>,
<2528,2527,2477>,
<2477,2478,2528>,
<2529,2528,2478>,
<2478,2479,2529>,
<2530,2529,2479>,
<2479,2480,2530>,
<2531,2530,2480>,
<2480,2481,2531>,
<2532,2531,2481>,
<2481,2482,2532>,
<2533,2532,2482>,
<2482,2483,2533>,
<2534,2533,2483>,
<2483,2484,2534>,
<2535,2534,2484>,
<2484,2485,2535>,
<2536,2535,2485>,
<2485,2486,2536>,
<2537,2536,2486>,
<2486,2487,2537>,
<2538,2537,2487>,

<2487,2488,2538>,
<2539,2538,2488>,
<2488,2489,2539>,
<2540,2539,2489>,
<2489,2490,2540>,
<2541,2540,2490>,
<2490,2491,2541>,
<2542,2541,2491>,
<2491,2492,2542>,
<2543,2542,2492>,
<2492,2493,2543>,
<2544,2543,2493>,
<2493,2494,2544>,
<2545,2544,2494>,
<2494,2495,2545>,
<2546,2545,2495>,
<2495,2496,2546>,
<2547,2546,2496>,
<2496,2497,2547>,
<2548,2547,2497>,
<2497,2498,2548>,
<2549,2548,2498>,
<2499,2500,2550>,
<2551,2550,2500>,
<2500,2501,2551>,
<2552,2551,2501>,
<2501,2502,2552>,
<2553,2552,2502>,
<2502,2503,2553>,
<2554,2553,2503>,
<2503,2504,2554>,
<2555,2554,2504>,
<2504,2505,2555>,
<2556,2555,2505>,
<2505,2506,2556>,
<2557,2556,2506>,
<2506,2507,2557>,
<2558,2557,2507>,
<2507,2508,2558>,
<2559,2558,2508>,
<2508,2509,2559>,
<2560,2559,2509>,
<2509,2510,2560>,
<2561,2560,2510>,
<2510,2511,2561>,
<2562,2561,2511>,
<2511,2512,2562>,
<2563,2562,2512>,
<2512,2513,2563>,
<2564,2563,2513>,
<2513,2514,2564>,
<2565,2564,2514>,
<2514,2515,2565>,
<2566,2565,2515>,
<2515,2516,2566>,
<2567,2566,2516>,

<2516,2517,2567>,
<2568,2567,2517>,
<2517,2518,2568>,
<2569,2568,2518>,
<2518,2519,2569>,
<2570,2569,2519>,
<2519,2520,2570>,
<2571,2570,2520>,
<2520,2521,2571>,
<2572,2571,2521>,
<2521,2522,2572>,
<2573,2572,2522>,
<2522,2523,2573>,
<2574,2573,2523>,
<2523,2524,2574>,
<2575,2574,2524>,
<2524,2525,2575>,
<2576,2575,2525>,
<2525,2526,2576>,
<2577,2576,2526>,
<2526,2527,2577>,
<2578,2577,2527>,
<2527,2528,2578>,
<2579,2578,2528>,
<2528,2529,2579>,
<2580,2579,2529>,
<2529,2530,2580>,
<2581,2580,2530>,
<2530,2531,2581>,
<2582,2581,2531>,
<2531,2532,2582>,
<2583,2582,2532>,
<2532,2533,2583>,
<2584,2583,2533>,
<2533,2534,2584>,
<2585,2584,2534>,
<2534,2535,2585>,
<2586,2585,2535>,
<2535,2536,2586>,
<2587,2586,2536>,
<2536,2537,2587>,
<2588,2587,2537>,
<2537,2538,2588>,
<2589,2588,2538>,
<2538,2539,2589>,
<2590,2589,2539>,
<2539,2540,2590>,
<2591,2590,2540>,
<2540,2541,2591>,
<2592,2591,2541>,
<2541,2542,2592>,
<2593,2592,2542>,
<2542,2543,2593>,
<2594,2593,2543>,
<2543,2544,2594>,
<2595,2594,2544>,

```

<2544,2545,2595>,
<2596,2595,2545>,
<2545,2546,2596>,
<2597,2596,2546>,
<2546,2547,2597>,
<2598,2597,2547>,
<2547,2548,2598>,
<2599,2598,2548>,
<2548,2549,2599>,
<2600,2599,2549>,
}

}

```

Sekarang kita mendefinisikan dua cakram, yang akan berpotongan dengan permukaan.

```

>cl=povdisc([0.5,0.5,0],[1,1,0],2); ...
>ll=povdisc([0,0,1/4],[0,0,1],2);

```

Tulis permukaan dikurangi dua cakram.

```

>writeln(povdifference(mesh,povunion([cl,ll]),povlook(green)));

```

```

difference {
object { mesh1 }
union { cylinder { <0.492929,0.492929,0>, <0.507071,0.507071,0>, 2
texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
finish { ambient 0.2 }
}
cylinder { <0,0,0.24>, <0,0,0.26>, 2
texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
finish { ambient 0.2 }
}
texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
finish { ambient 0.2 }
}
texture { pigment { color rgb <0.0627451,0.564706,0.0627451> } }
finish { ambient 0.2 }
}

```

Tulis dua persimpangan.

```

>writeln(povintersection([mesh,cl],povlook(red))); ...
>writeln(povintersection([mesh,ll],povlook(gray)));

```

```

intersection { object { mesh1 }
cylinder { <0.492929,0.492929,0>, <0.507071,0.507071,0>, 2
texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
finish { ambient 0.2 }
}
texture { pigment { color rgb <0.564706,0.0627451,0.0627451> } }
finish { ambient 0.2 }
}

```

```

intersection { object { mesh1 }
cylinder { <0,0,0.24>, <0,0,0.26>, 2
texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
finish { ambient 0.2 }
}
texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
finish { ambient 0.2 }
}

```

Write a point at the maximum.

```
>writeln(povpoint([1/2,1/2,1/4],povlook(gray),size=2*defaultpointsize));
```

```

sphere { <0.5,0.5,0.25>, 0.04 texture { pigment { color rgb <0.470588,0.470588,0.470588>
finish { ambient 0.2 }

}

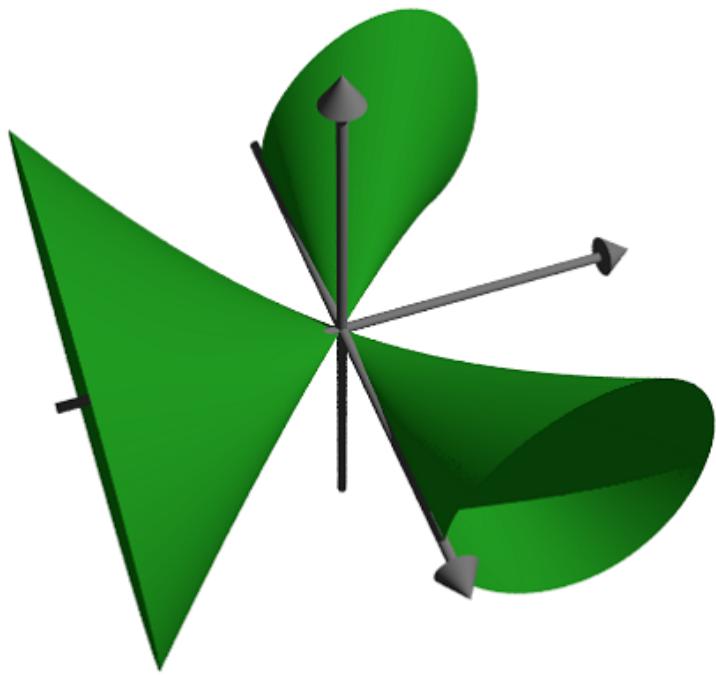
```

Tambahkan sumbu dan selesaikan.

```
>writeAxes(0,1,0,1,0,1,d=0.015); ...
>povend();
```

```

union {
    cylinder { <-0.1,0,0>,<1.1,0,0>,0.015 }
    cone {
        <1.19,0,0>,0
        <1.1,0,0>,0.06
    }
    texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
}
union {
    cylinder { <-0.1,0,0>,<1.1,0,0>,0.015 }
    cone {
        <1.19,0,0>,0
        <1.1,0,0>,0.06
    }
    rotate 90*z
    texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
}
union {
    cylinder { <-0.1,0,0>,<1.1,0,0>,0.015 }
    cone {
        <1.19,0,0>,0
        <1.1,0,0>,0.06
    }
    rotate -90*y
    texture { pigment { color rgb <0.470588,0.470588,0.470588> } }
}
```



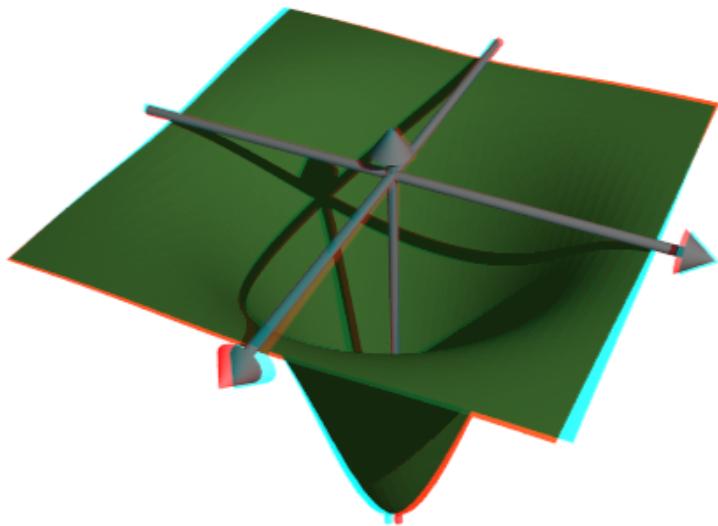
Anaglyph di Povray

Untuk menghasilkan anaglyph untuk kacamata merah/sian, Povray harus berjalan dua kali dari posisi kamera yang berbeda. Ini menghasilkan dua file Povray dan dua file PNG, yang dimuat dengan fungsi loadanaglyph().

Tentu saja, Anda memerlukan kacamata merah/sian untuk melihat contoh berikut dengan benar.

Fungsi pov3d() memiliki sakelar sederhana untuk menghasilkan anaglyphs.

```
>pov3d("-exp(-x^2-y^2)/2",r=2,height=45°,>anaglyph, ...
>    center=[0,0,0.5],zoom=3.5);
```



Jika Anda membuat adegan dengan objek, Anda perlu menempatkan generasi adegan ke dalam fungsi, dan menjalankannya dua kali dengan nilai yang berbeda untuk parameter anaglyph.

```
>function myscene ...
s=povsphere(povc,1);
cl=povcylinder(-povz,povz,0.5);
clk=povobject(cl,rotate=xrotate(90°));
cly=povobject(cl,rotate=yrotate(90°));
c=povbox([-1,-1,0],1);
un=povunion([cl,clk,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);
```

Mendefinisikan Objek sendiri

Antarmuka povray Euler berisi banyak objek. Tapi Anda tidak terbatas pada ini. Anda dapat membuat objek sendiri, yang menggabungkan objek lain, atau objek yang sama sekali baru.

Kami mendemonstrasikan sebuah torus. Perintah Povray untuk ini adalah "torus". Jadi kami mengembalikan string dengan perintah ini dan parameternya. Perhatikan bahwa torus selalu berpusat di titik asal.

```
>function povdonat (r1,r2,look "") ...
```

```
    return "torus {" + r1 + "," + r2 + look + "}";  
endfunction
```

Inilah torus pertama kita

```
>t1=povdonat(0.8,0.2)
```

```
torus {0.8,0.2}
```

Mari kita gunakan objek ini untuk membuat torus kedua, diterjemahkan dan diputar.

```
>t2=povobject(t1,rotate=xrotate(90°),translate=[0.8,0,0])
```

```
object { torus {0.8,0.2}  
rotate 90 *x  
translate <0.8,0,0>  
}
```

Sekarang kita menempatkan objek-objek ini ke dalam sebuah adegan. Untuk tampilan, kami menggunakan Phong Shading.

```
>povstart(center=[0.4,0,0],angle=0°,zoom=3.8,aspect=1.5); ...  
>writeln(povobject(t1,povlook(green,phong=1))); ...  
>writeln(povobject(t2,povlook(green,phong=1)));  
>povstart(center=[0.4,0,0],angle=0°,zoom=3.8,aspect=1.5); ...  
>writeln(povobject(t1,povlook(green,phong=1))); ...  
>writeln(povobject(t2,povlook(red,phong=1))); ...  
> ...
```

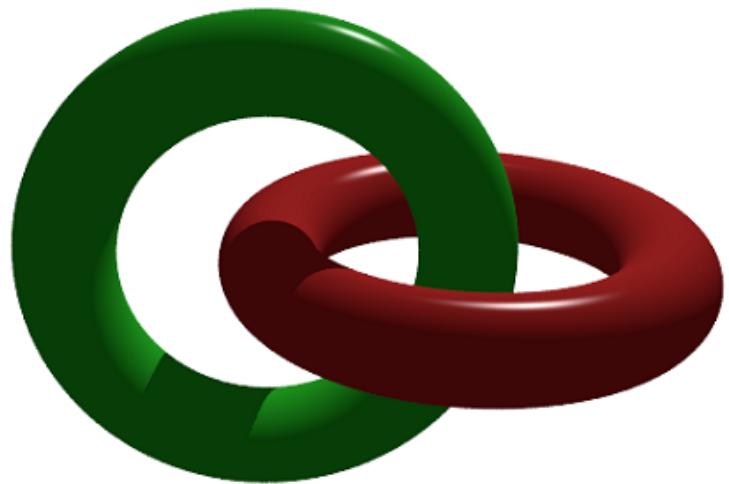
```
>povend();
```

memanggil program Povray. Namun, jika terjadi kesalahan, itu tidak menampilkan kesalahan. Karena itu Anda harus menggunakan

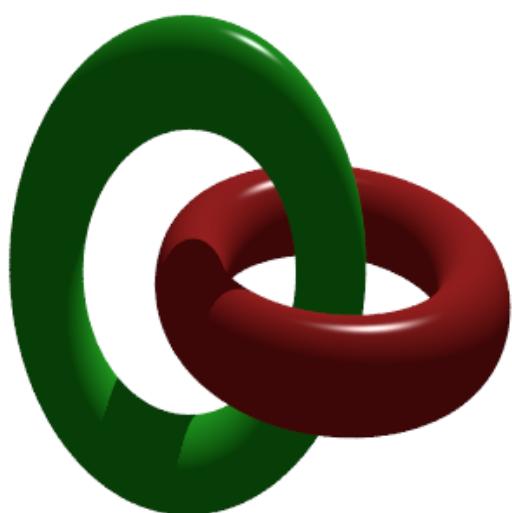
```
>povend(<keluar>);
```

jika ada yang tidak berhasil. Ini akan membiarkan jendela Povray terbuka.

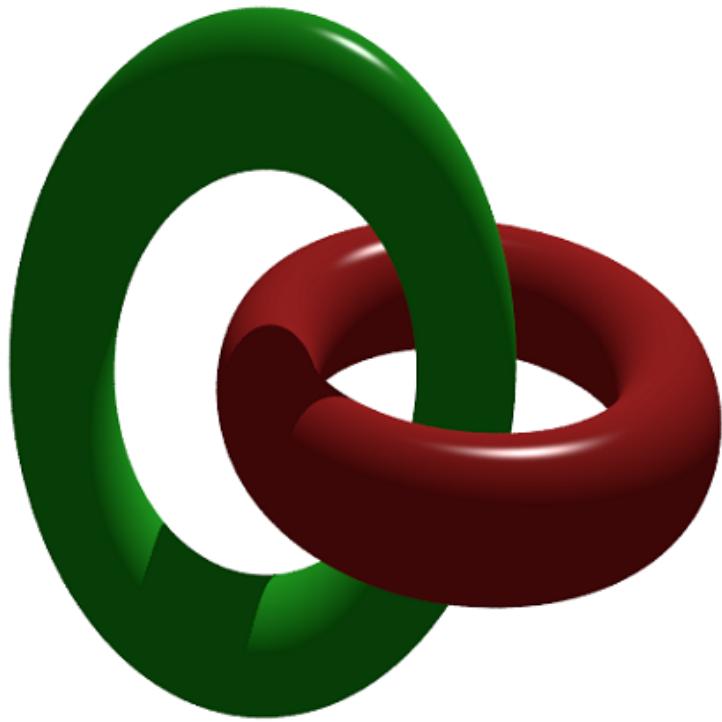
```
>povend(h=320,w=480);
```



```
>povend (h=320,w=320);
```



```
>povend (h=480,w=480);
```



```
>povend(h=100,w=120);
```



```
>povend(h=120,w=80);
```

Berikut adalah contoh yang lebih rumit. Kami memecahkan

$$Ax \leq b, \quad x \geq 0, \quad c.x \rightarrow \text{Max.}$$

dan menunjukkan titik layak dan optimal dalam plot 3D.

```
>A=[10,8,4;5,6,8;6,3,2;9,5,6];
>b=[10,10,10,10]';
>c=[1,1,1];
```

Pertama, mari kita periksa, apakah contoh ini memiliki solusi sama sekali.

```
>x=simplex(A,b,c,>max,>check)'
```

```
[0,    1,    0.5]
```

Ya, sudah.

Selanjutnya kita mendefinisikan dua objek. Yang pertama adalah pesawat

$$a \cdot x \leq b$$

```
>function oneplane (a,b,look="") ...
```

```
    return povplane(a,b,look)
endfunction
```

Kemudian kita mendefinisikan persimpangan dari semua setengah ruang dan sebuah kubus.

```
>function adm (A, b, r, look="") ...
```

```
    ol=[];
    loop 1 to rows(A); ol=ol|oneplane(A[#],b[#]); end;
    ol=ol|povbox([0,0,0],[r,r,r]);
    return povintersection(ol,look);
endfunction
```

Kita sekarang dapat merencanakan adegannya.

```
>povstart(angle=120°,center=[0.5,0.5,0.5],zoom=3.5); ...
>writeln(adm(A,b,2,povlook(green,0.4))); ...
>writeAxes(0,1.3,0,1.6,0,1.5); ...
```

Berikut ini adalah lingkaran di sekitar optimal.

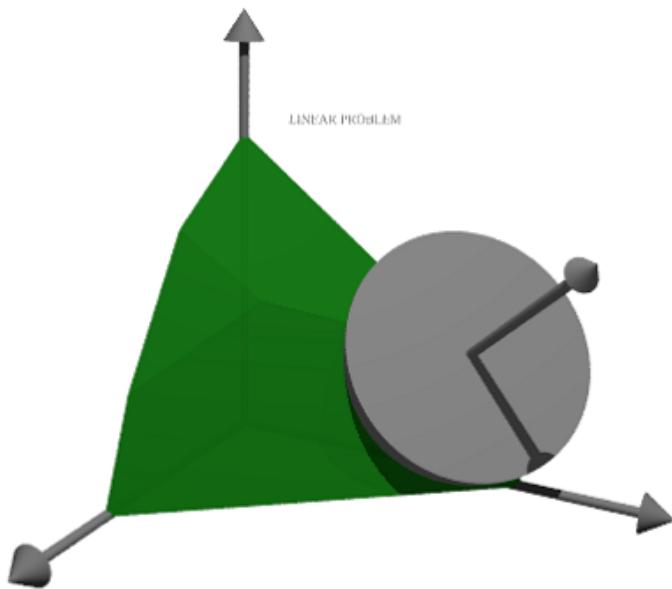
```
>writeln(povintersection([povsphere(x,0.5),povplane(c,c.x')], ...
>    povlook(red,0.9)));
```

Dan kesalahan ke arah yang optimal.

```
>writeln(povarrow(x,c*0.5,povlook(red)));
```

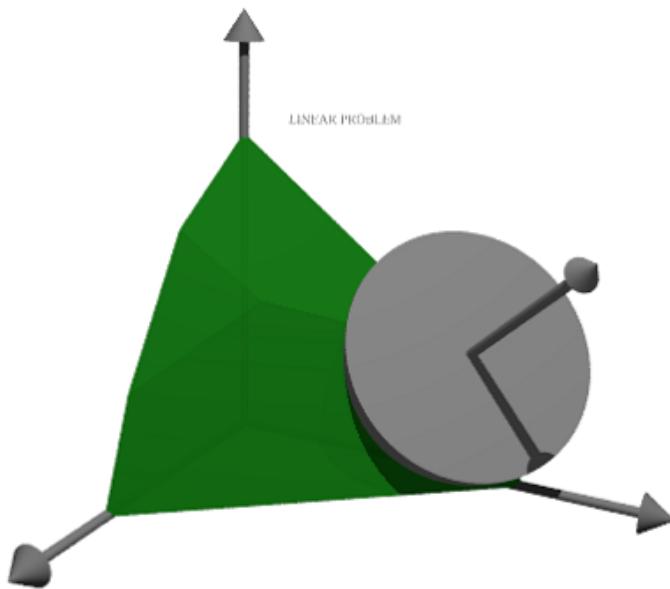
Kami menambahkan teks ke layar. Teks hanyalah objek 3D. Kita perlu menempatkan dan memutarinya menuarut pandangan kita.

```
>writeln(povtext("LINEAR PROBLEM", [0,0.2,1.3], size=0.05, rotate=125°)); ...
>povend();
```



```
>writeln(povtext("LINEAR PROBLEM", [0,0.5,1.3],size=0.01,rotate=360°)); ...
>povend();
```

```
object { text { ttf "timrom.ttf" "LINEAR PROBLEM" 0.04 ,0  texture { pigment { color rgb
  finish { ambient 0.2 }
}
scale 0.01
rotate 90 *x  rotate 360 *z
translate <0,0.5,1.3>
}
```



Lebih Banyak Contoh

Anda dapat menemukan beberapa contoh lagi untuk Povray di Euler di file berikut.

Lihat: Examples/Dandelin Spheres

Lihat: Examples/Donut Math

Lihat: Examples/Trefoil Knot

Lihat: Examples/Optimization by Affine Scaling

BAB 5

KB PEKAN 6-7: MENGGUNAKAN EMT UNTUK KALKULUS

article
eumat

Kalkulus dengan EMT

Shalih Abdillah
22305144009
Matematika B 2022

Materi Kalkulus mencakup di antaranya:

- Fungsi (fungsi aljabar, trigonometri, eksponensial, logaritma, komposisi fungsi)
- Limit Fungsi,
- Turunan Fungsi,
- Integral Tak Tentu,
- Integral Tentu dan Aplikasinya,
- Barisan dan Deret (kekonvergenan barisan dan deret).

EMT (bersama Maxima) dapat digunakan untuk melakukan semua perhitungan di dalam kalkulus, baik secara numerik maupun analitik (eksak).

Mendefinisikan Fungsi

Terdapat beberapa cara mendefinisikan fungsi pada EMT, yakni:

- Menggunakan format `nama_fungsi := rumus fungsi` (untuk fungsi numerik),
- Menggunakan format `nama_fungsi &= rumus fungsi` (untuk fungsi simbolik, namun dapat dihitung secara numerik),
- Menggunakan format `nama_fungsi &&= rumus fungsi` (untuk fungsi simbolik murni, tidak dapat dihitung langsung),
- Fungsi sebagai program EMT.

Setiap format harus diawali dengan perintah `function` (bukan sebagai ekspresi).

Berikut adalah beberapa contoh cara mendefinisikan fungsi:

$$f(x) = 2x^2 + e^{\sin(x)}.$$

```
>function f(x) := 2*x^2+exp(sin(x)) // fungsi numerik  
>f(0), f(1), f(pi)
```

```
1  
4.31977682472  
20.7392088022
```

```
>f(a) // tidak dapat dihitung nilainya
```

Variable or function a not found.
Error in:
f(a) // tidak dapat dihitung nilainya ...
^

Silakan Anda plot kurva fungsi di atas!

Berikutnya kita definisikan fungsi:

$$g(x) = \frac{\sqrt{x^2 - 3x}}{x + 1}.$$

```
>function g(x) := sqrt(x^2-3*x) / (x+1)  
>g(3)
```

```
0
```

```
>g(0)
```

```
0
```

```
>g(1) // kompleks, tidak dapat dihitung oleh fungsi numerik
```

Floating point error!
Error in sqrt
Try "trace errors" to inspect local variables after errors.
g:
useglobal; return sqrt(x^2-3*x) / (x+1)
Error in:
g(1) // kompleks, tidak dapat dihitung oleh fungsi numerik ...
^

Silakan Anda plot kurva fungsi di atas!

```
>f(g(5)) // komposisi fungsi
```

```
2.20920171961
```

```
>g(f(5))
```

```
0.950898070639
```

```
>function h(x) := f(g(x)) // definisi komposisi fungsi  
>h(5) // sama dengan f(g(5))
```

```
2.20920171961
```

Silakan Anda plot kurva fungsi komposisi fungsi f dan g:

$$h(x) = f(g(x))$$

dan

$$u(x) = g(f(x))$$

bersama-sama kurva fungsi f dan g dalam satu bidang koordinat.

```
>f(0:10) // nilai-nilai f(0), f(1), f(2), ..., f(10)
```

```
[1, 4.31978, 10.4826, 19.1516, 32.4692, 50.3833, 72.7562,  
99.929, 130.69, 163.51, 200.58]
```

```
>fmap(0:10) // sama dengan f(0:10), berlaku untuk semua fungsi
```

```
[1, 4.31978, 10.4826, 19.1516, 32.4692, 50.3833, 72.7562,  
99.929, 130.69, 163.51, 200.58]
```

```
>gmap(200:210)
```

```
[0.987534, 0.987596, 0.987657, 0.987718, 0.987778, 0.987837,  
0.987896, 0.987954, 0.988012, 0.988069, 0.988126]
```

Misalkan kita akan mendefinisikan fungsi

$$f(x) = \begin{cases} x^3 & x > 0 \\ x^2 & x \leq 0. \end{cases}$$

Fungsi tersebut tidak dapat didefinisikan sebagai fungsi numerik secara "inline" menggunakan format `:=`, melainkan didefinisikan sebagai program. Perhatikan, kata "map" digunakan agar fungsi dapat menerima vektor sebagai input, dan hasilnya berupa vektor. Jika tanpa kata "map" fungsinya hanya dapat menerima input satu nilai.

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

```
if x>0 then return x^3  
else return x^2  
endif;  
endfunction
```

```
>f(1)
```

1

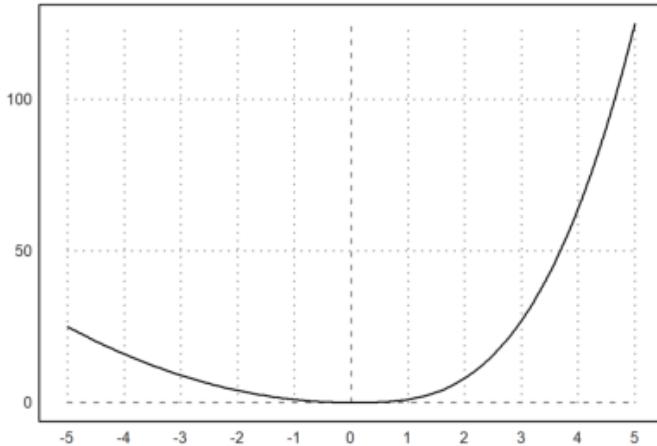
```
>f(-2)
```

4

```
>f(-5:5)
```

[25, 16, 9, 4, 1, 0, 1, 8, 27, 64, 125]

```
>aspect(1.5); plot2d("f(x)", -5, 5):
```



```
>function f(x) &= 2*x // fungsi simbolik
```

$$2^x$$

```
>$f(a) // nilai fungsi secara simbolik
```

$$2e^a$$

```
>f(E) // nilai fungsi berupa bilangan desimal
```

30.308524483

```
>$f(E), $float(%)
```

30.30852448295852

images/Shalih Abdillah_22305144009_ApKom5-009-lar

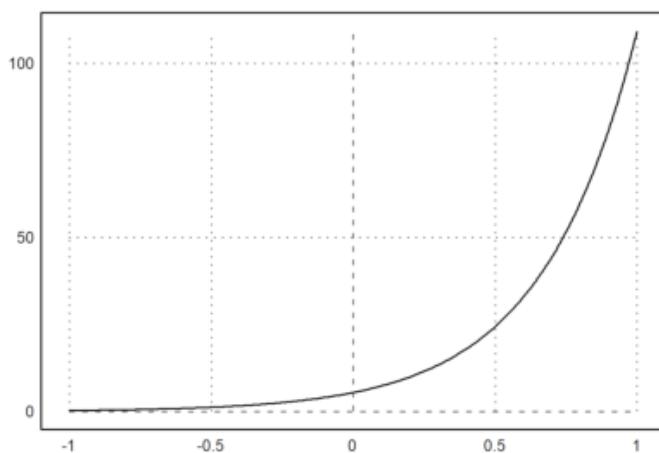
```
>function g(x) &= 3*x+1
```

3 x + 1

```
>function h(x) &= f(g(x)) // komposisi fungsi
```

3 x + 1
2 E

```
>plot2d("h(x)", -1, 1):
```



Latihan

Bukalah buku Kalkulus. Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan fungsi-fungsi tersebut dan komposisinya di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi). Untuk setiap fungsi, hitung beberapa nilainya, baik untuk satu nilai maupun vektor. Gambar grafik fungsi-fungsi tersebut dan komposisi-komposisi 2 fungsi.

Juga, carilah fungsi beberapa (dua) variabel. Lakukan hal sama seperti di atas.

Soal 1

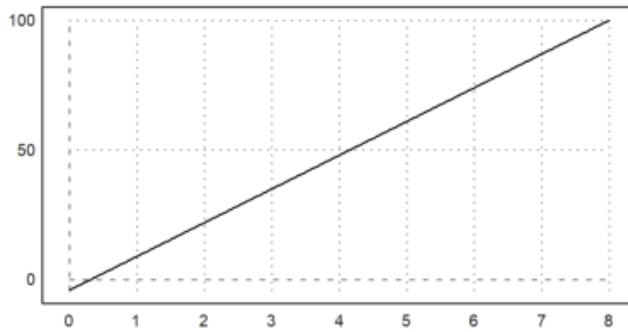
```
>function f(x) := 3*x + 10*x - 4  
>f(3)
```

35

```
>f(0:8)
```

[-4, 9, 22, 35, 48, 61, 74, 87, 100]

```
>aspect(2); plot2d("f(x)", 0, 8):
```



Soal 2

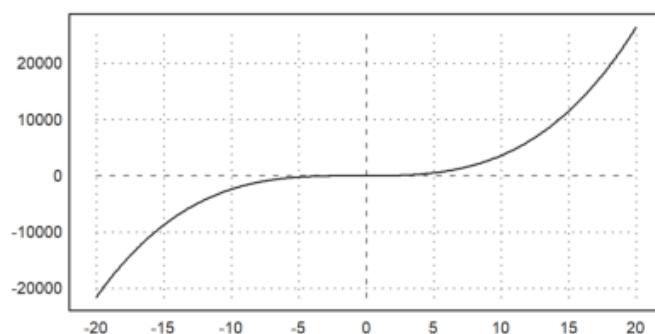
```
>function f(y) := 6*y^2 + 3*y^3 - 2  
>f(3)
```

133

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

[7, 46, 133, 286, 523]

```
> aspect(2); plot2d("f(x)", -20, 20):
```



Soal 3

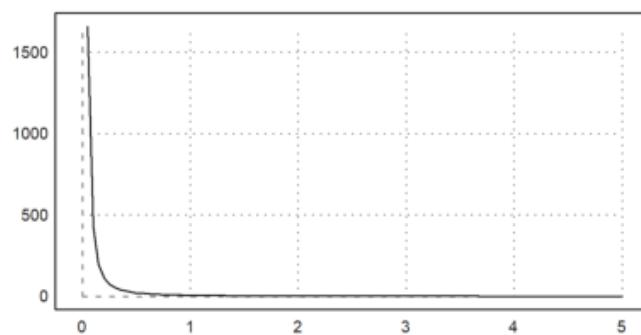
```
> function f(x) := (3*x+4) / (x^2)
> f(4)
```

1

```
> f(1:10)
```

```
[7, 2.5, 1.44444, 1, 0.76, 0.611111, 0.510204, 0.4375,
0.382716, 0.34]
```

```
> aspect(2); plot2d("f(x)", 0, 5):
```



Soal 4

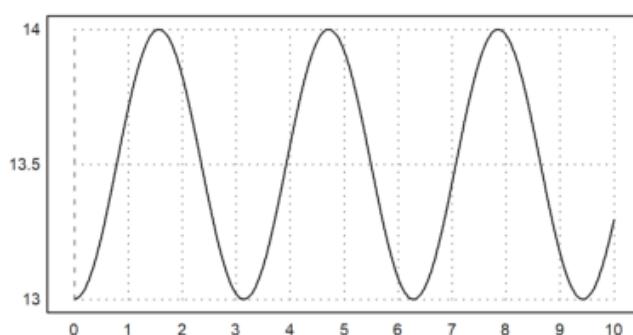
```
> function f(x) := sin(x)^2 + 13
> f(30)
```

13.9762064902

```
> f(0)
```

13

```
> aspect(2); plot2d("f(x)", 0, 10):
```



Soal 5

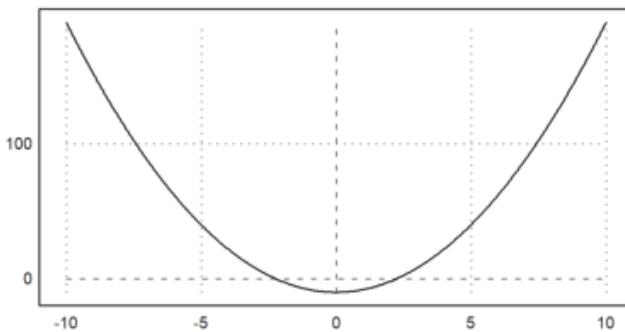
```
> function f(x,y) := -x^2 + 3*y^2 -10  
> f(2,5)
```

61

```
> f(0:10,0:10)
```

[-10, -8, -2, 8, 22, 40, 62, 88, 118, 152, 190]

```
> aspect(2); plot2d("f(x,x)",-10,10):
```



Soal 6

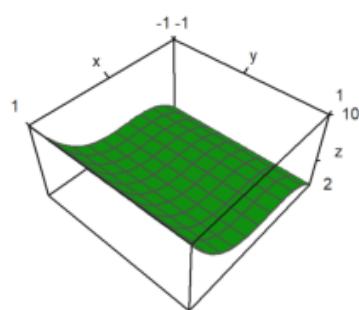
```
> function f(x,y,z) := x^3 + 2*y^2 + 3*z^3 + 4  
> f(1,2,3)
```

94

```
> f(3,4,5)
```

438

```
> aspect(2); plot3d("f(x,x,x)",angle=40, height=45°):
```



Menghitung Limit

Perhitungan limit pada EMT dapat dilakukan dengan menggunakan fungsi Maxima, yakni "limit". Fungsi "limit" dapat digunakan untuk menghitung limit fungsi dalam bentuk ekspresi maupun fungsi yang sudah didefinisikan sebelumnya. Nilai limit dapat dihitung pada sebarang nilai atau pada tak hingga (-inf, minf, dan inf). Limit kiri dan limit kanan juga dapat dihitung, dengan cara memberi opsi "plus" atau "minus". Hasil limit dapat berupa nilai, "und" (tak definisi), "ind" (tak tentu namun terbatas), "infinity" (kompleks tak hingga). Perhatikan beberapa contoh berikut. Perhatikan cara menampilkan perhitungan secara lengkap, tidak hanya menampilkan hasilnya saja.

```
>$showev('limit(sqrt(x^2-3*x)/(x+1),x,inf))
```

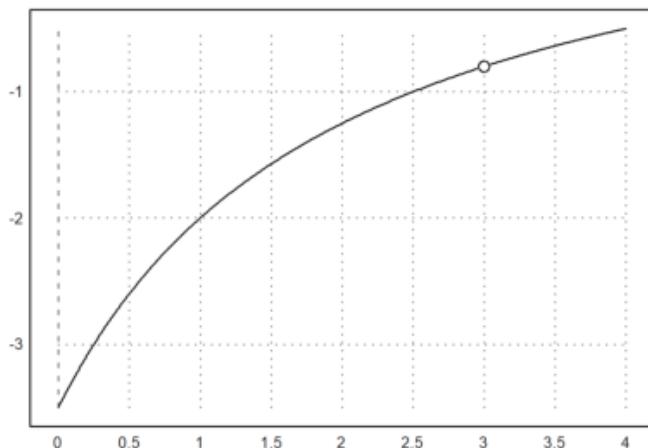
```
Maxima said:  
limit: variable must be a symbol or subscripted symbol; found:  
errexpl  
#0: showev(f='limit([0,sqrt(-4.999975000072321e-7*r+2.7777500001498e-14*r^2)/(1+1.666658  
-- an error. To debug this try: debugmode(true);  
  
Error in:  
$showev('limit(sqrt(x^2-3*x)/(x+1),x,inf)) ...  
^
```

```
>$limit((x^3-13*x^2+51*x-63)/(x^3-4*x^2-3*x+18),x,3)
```

$$-\frac{4}{5}$$

maxima: limit((x^3-13*x^2+51*x-63)/(x^3-4*x^2-3*x+18),x,3)=limit((x^3-13*x^2+51*x-63)/(x^3-4*x^2-3*x+18),x,3)
Fungsi tersebut diskontinu di titik $x=3$. Berikut adalah grafik fungsinya.

```
>aspect(1.5); plot2d("(x^3-13*x^2+51*x-63)/(x^3-4*x^2-3*x+18)",0,4); plot2d(3,-4/5,>points
```

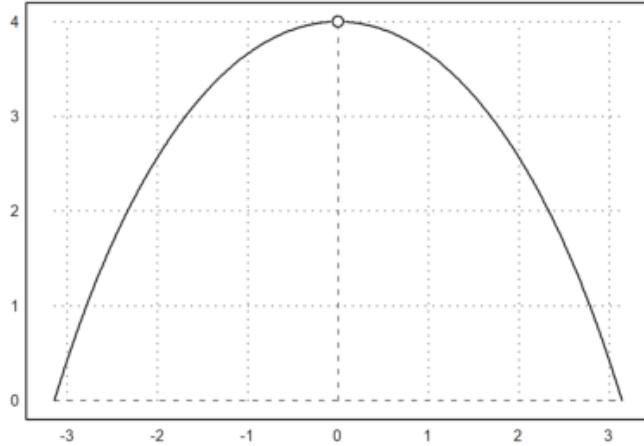


```
>$limit(2*x*sin(x)/(1-cos(x)),x,0)
```

maxima: 'limit(2*x*sin(x)/(1-cos(x)),x,0)

Fungsi tersebut diskontinu di titik $x=0$. Berikut adalah grafik fungsinya.

```
>plot2d("2*x*sin(x) / (1-cos(x))", -pi, pi); plot2d(0, 4, >points, style="ow", >add) :
```



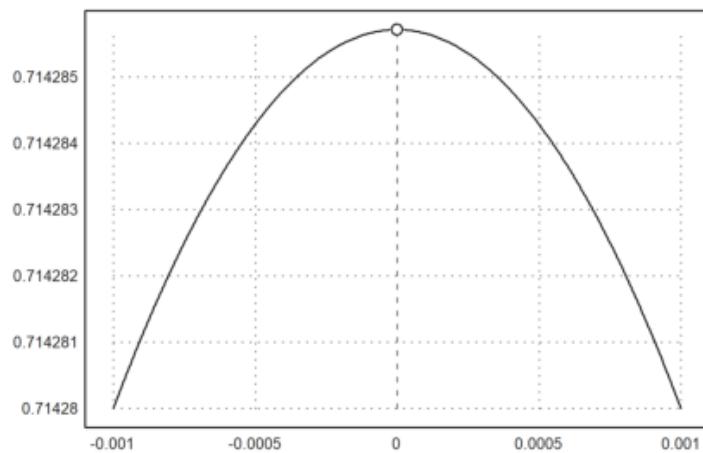
```
>$limit(cot(7*h)/cot(5*h), h, 0)
```

$$\frac{5}{7}$$

$$\lim_{h \rightarrow 0} \frac{\cot(7h)}{\cot(5h)} = \frac{5}{7}$$

Fungsi tersebut juga diskontinu (karena tidak terdefinisi) di $x=0$. Berikut adalah grafiknya.

```
>plot2d("cot(7*x)/cot(5*x)", -0.001, 0.001); plot2d(0, 5/7, >points, style="ow", >add) :
```

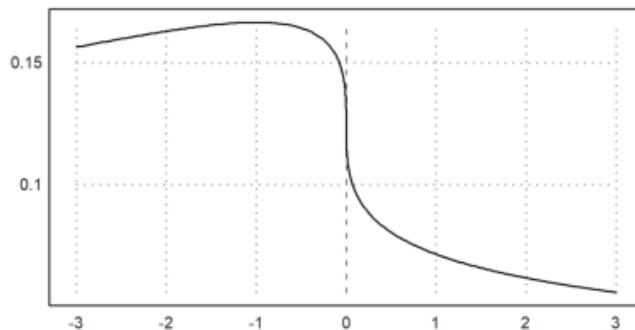


```
>$showev('limit(((x/8)^(1/3)-1)/(x-8),x,8))
```

$$\lim_{x \rightarrow 8} \frac{\frac{x^{\frac{1}{3}}}{2} - 1}{x - 8} = \frac{1}{24}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>plot2d("((x/8)^(1/3)-1)/(x-8)",-3,3); plot2d(0,5/7,>points,style="ow",>add):
```

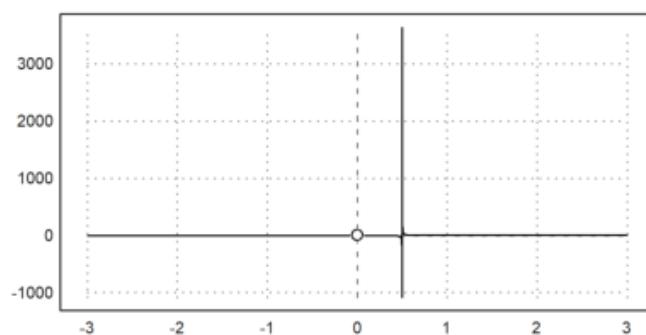


```
>$showev('limit(1/(2*x-1),x,0))
```

$$\lim_{x \rightarrow 0} \frac{1}{2x - 1} = -1$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>plot2d("1/(2*x-1)",-3,3); plot2d(0,5/7,>points,style="ow",>add):
```

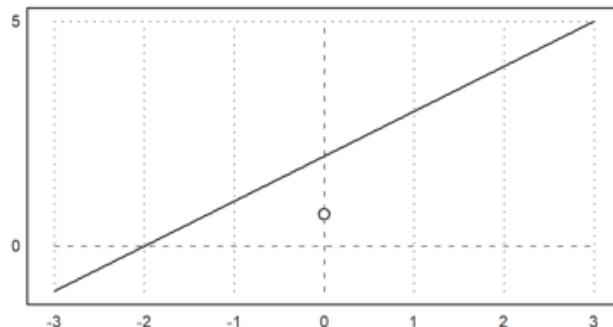


```
>$showev('limit((x^2-3*x-10)/(x-5),x,5))
```

$$\lim_{x \rightarrow 5} \frac{x^2 - 3x - 10}{x - 5} = 7$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>plot2d("(x^2-3*x-10)/(x-5)", -3, 3); plot2d(0, 5/7, >points, style="ow", >add):
```

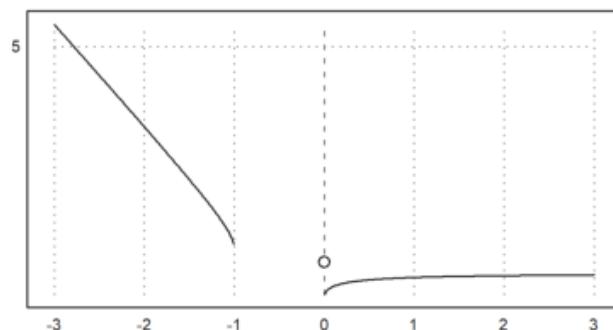


```
>$showev('limit(sqrt(x^2+x)-x, x, inf))
```

$$\lim_{x \rightarrow \infty} \sqrt{x^2 + x} - x = \frac{1}{2}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>plot2d("sqrt(x^2+x)-x", -3, 3); plot2d(0, 5/7, >points, style="ow", >add):
```



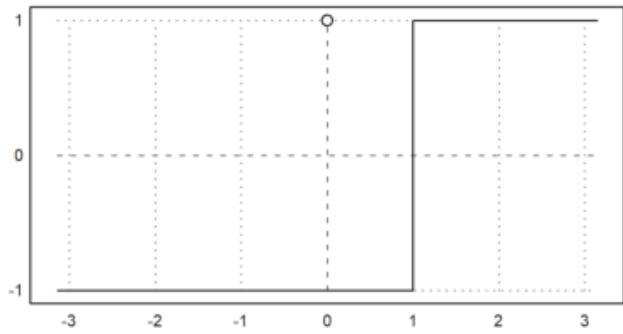
```
>$showev('limit(abs(x-1)/(x-1), x, 1, minus))
```

$$\lim_{x \uparrow 1} \frac{|x - 1|}{x - 1} = -1$$

Hitung limit di atas untuk x menuju 1 dari kanan.

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

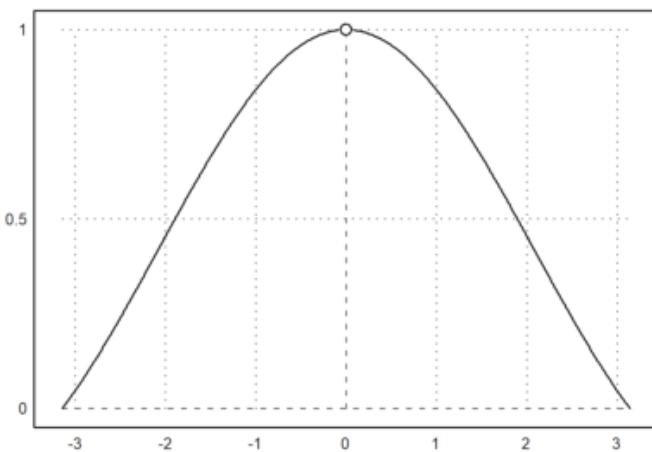
```
>plot2d("abs(x-1)/(x-1)", -pi, pi); plot2d(0, 1, >points, style="ow", >add):
```



```
>$showev('limit(sin(x)/x,x,0))
```

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

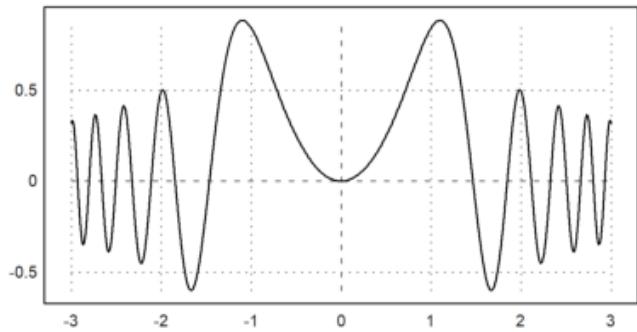
```
>plot2d("sin(x)/x",-pi,pi); plot2d(0,1,>points,style="ow",>add):
```



```
>$showev('limit(sin(x^3)/x,x,0))
```

$$\lim_{x \rightarrow 0} \frac{\sin x^3}{x} = 0$$

```
>plot2d("sin(x^3)/x",-3,3); plot2d(0,1,>points,style="ow",>add):
```



Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>$showev('limit(log(x), x, minf))
```

$$\lim_{x \rightarrow -\infty} \log x = \text{infinity}$$

```
>$showev('limit((-2)^x, x, inf))
```

$$\lim_{x \rightarrow \infty} (-2)^x = \text{infinity}$$

```
>$showev('limit(t-sqrt(2-t), t, 2, minus))
```

$$\lim_{t \uparrow 2} t - \sqrt{2-t} = 2$$

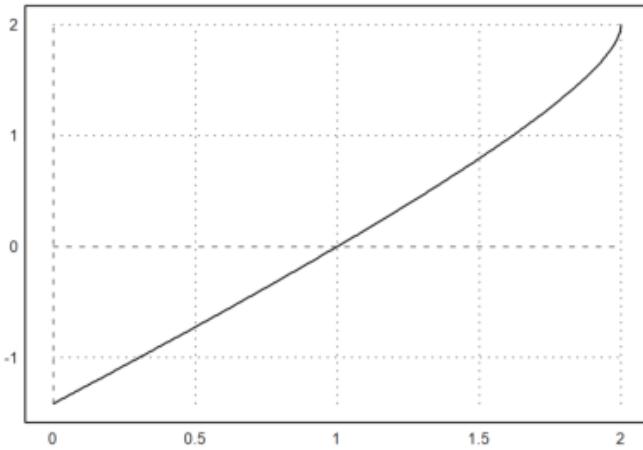
```
>$showev('limit(t-sqrt(2-t), t, 2, plus))
```

$$\lim_{t \downarrow 2} t - \sqrt{2-t} = 2$$

```
>$showev('limit(t-sqrt(2-t), t, 5, plus)) // Perhatikan hasilnya
```

$$\lim_{t \downarrow 5} t - \sqrt{2-t} = 5 - \sqrt{3}i$$

```
>plot2d("x-sqrt(2-x)", 0, 2):
```

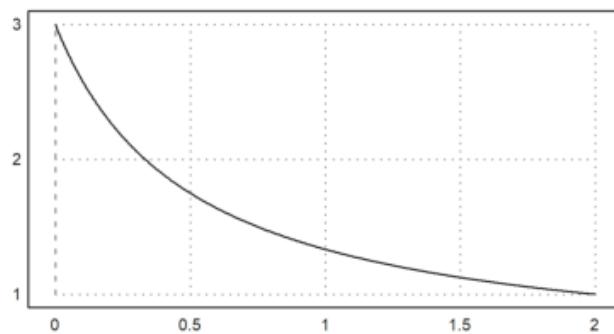


```
>$showev('limit((x^2-9)/(2*x^2-5*x-3), x, 3))
```

$$\lim_{x \rightarrow 3} \frac{x^2 - 9}{2x^2 - 5x - 3} = \frac{6}{7}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

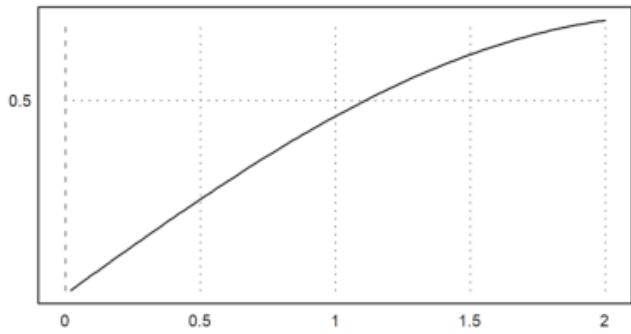
```
>plot2d("(x^2-9)/(2*x^2-5*x-3)", 0, 2):
```



```
>$showev('limit((1-cos(x))/x, x, 0))
```

$$\lim_{x \rightarrow 0} \frac{1 - \cos x}{x} = 0$$

```
>plot2d("(1-cos(x))/x", 0, 2):
```

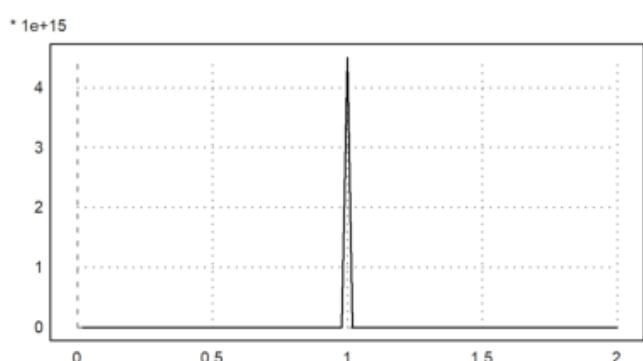


```
>$showev('limit((x^2+abs(x))/(x^2-abs(x)),x,0))
```

$$\lim_{x \rightarrow 0} \frac{|x| + x^2}{x^2 - |x|} = -1$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

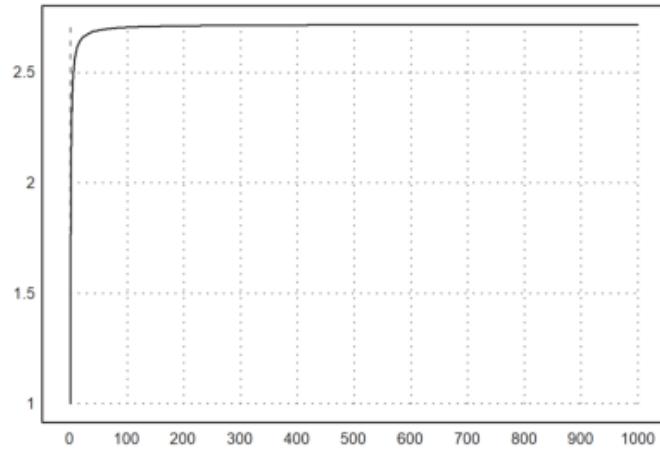
```
>plot2d("(x^2+abs(x))/(x^2-abs(x))", 0, 2) :
```



```
>$showev('limit((1+1/x)^x,x,inf))
```

$$\lim_{x \rightarrow \infty} \left(\frac{1}{x} + 1 \right)^x = e$$

```
>plot2d("(1+1/x)^x", 0, 1000) :
```



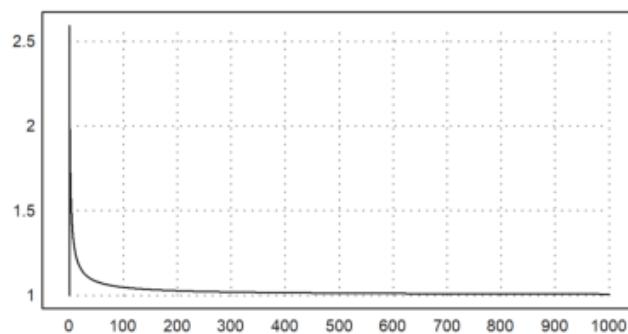
```
>$showev('limit((1+k/x)^x,x,inf))
```

$$\lim_{x \rightarrow \infty} \left(\frac{k}{x} + 1 \right)^x = e^k$$

```
>$showev('limit((1+x)^(1/x),x,0))
```

$$\lim_{x \rightarrow 0} (x + 1)^{\frac{1}{x}} = e$$

```
>plot2d("(1+x)^(1/x)",0,1000):
```



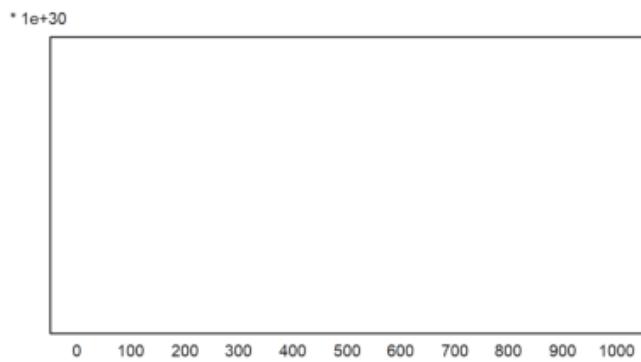
```
>$showev('limit((x/(x+k))^x,x,inf))
```

$$\lim_{x \rightarrow \infty} \left(\frac{x}{x + k} \right)^x = e^{-k}$$

```
>$showev('limit((E^x-E^2)/(x-2),x,2))
```

$$\lim_{x \rightarrow 2} \frac{e^x - e^2}{x - 2} = e^2$$

```
>plot2d(" (E^x-E^2) / (x-2) ",0,1000):
```



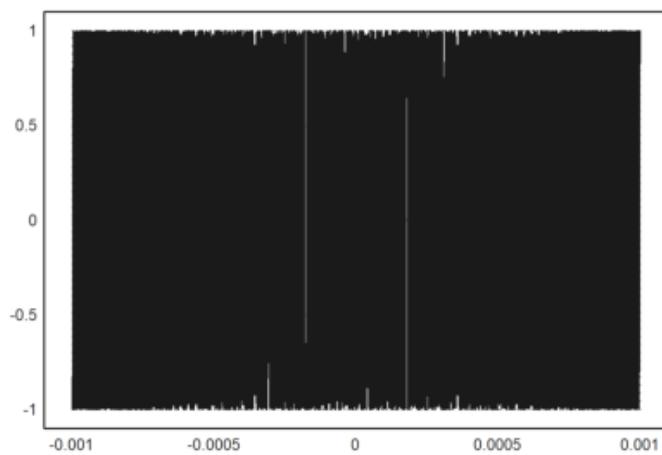
```
>$showev('limit(sin(1/x),x,0))
```

$$\lim_{x \rightarrow 0} \sin\left(\frac{1}{x}\right) = \text{ind}$$

```
>$showev('limit(sin(1/x),x,inf))
```

$$\lim_{x \rightarrow \infty} \sin\left(\frac{1}{x}\right) = 0$$

```
>plot2d("sin(1/x)",-0.001,0.001):
```



Latihan

Bukalah buku Kalkulus. Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi). Untuk setiap fungsi, hitung nilai limit fungsi tersebut di beberapa nilai dan di tak hingga. Gambar grafik fungsi tersebut untuk mengkonfirmasi nilai-nilai limit tersebut.

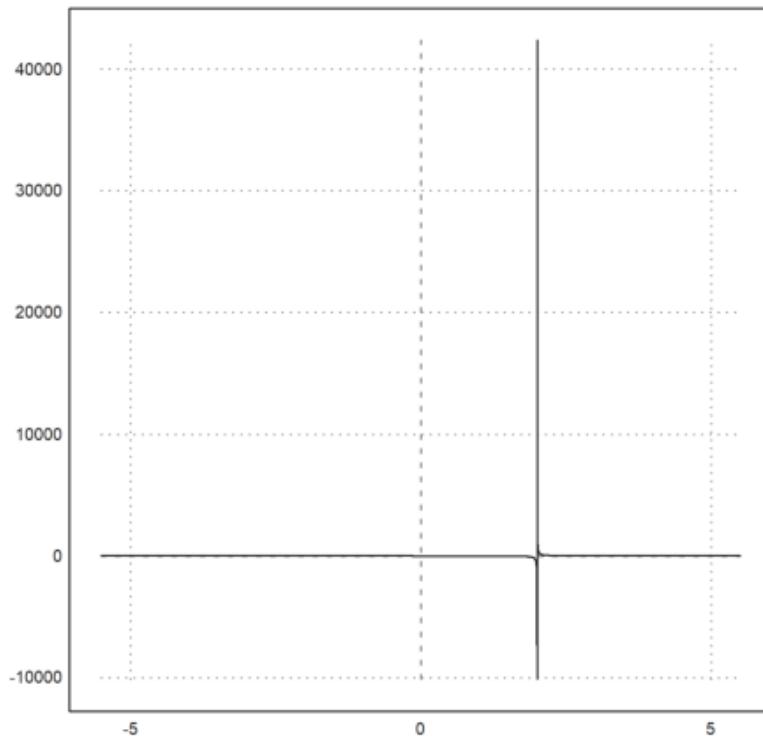
Fungsi 1

$$f(x) = \frac{8x - 7}{x - 2}$$

```
>showev('limit((8*x-7)/(x-2), x, 1))
```

$$\lim_{x \rightarrow 1} \frac{8x - 7}{x - 2} = -1$$

```
>plot2d("(8*x-7)/(x-2)", -5.5, 5.5):
```



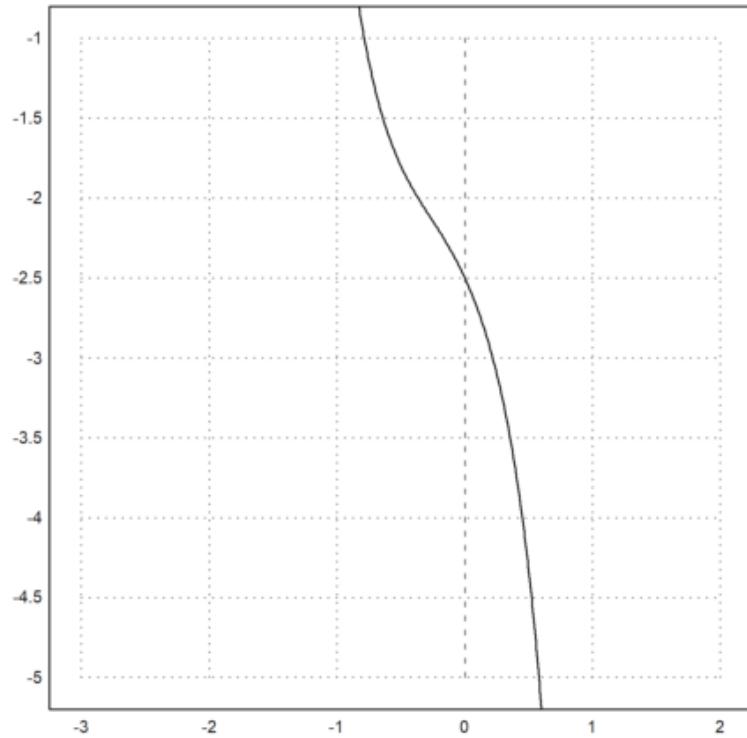
Fungsi 2

$$f(x) = \frac{4x^5 + 3x^3 + 2x^2 + x + 5}{x - 2}$$

```
>showev('limit((4*x^5+3*x^3+2*x^2+x+5)/(x-2), x, 1))
```

$$\lim_{x \rightarrow 1} \frac{4x^5 + 3x^3 + 2x^2 + x + 5}{x - 2} = -15$$

```
>plot2d("(4*x^5+3*x^3+2*x^2+x+5)/(x-2)", -3, 2, -5, -1):
```



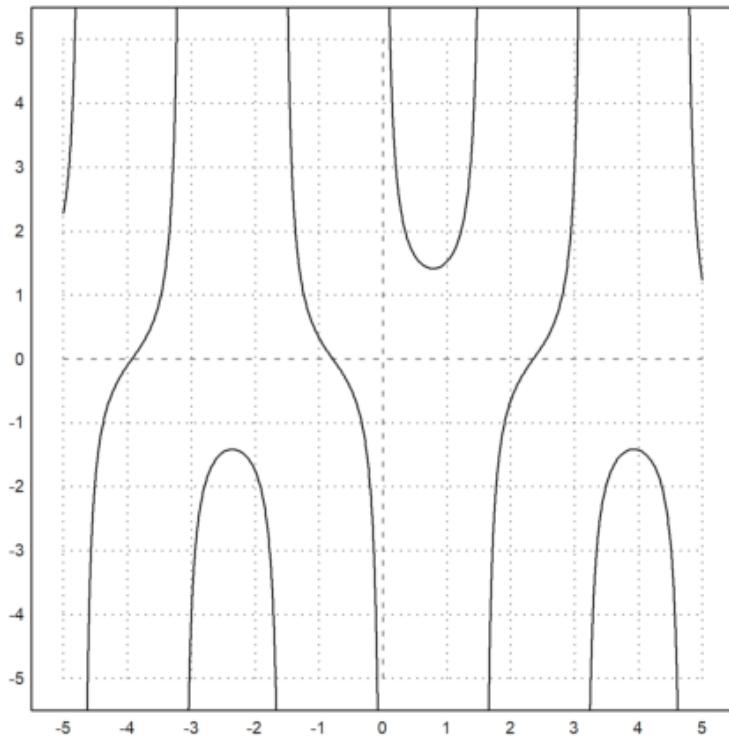
Fungsi 3

$$f(x) = \frac{\sin x + \cos x}{\sin(2x)}$$

```
>$showev('limit(((sin(x)+cos(x))/sin(2*x)),x,2))
```

$$\lim_{x \rightarrow 2} \frac{\sin x + \cos x}{\sin(2x)} = \frac{\sin 2}{\sin 4} + \frac{\cos 2}{\sin 4}$$

```
>plot2d("(sin(x)+cos(x))/(sin(2*x))", -5, 5, -5, 5):
```



Fungsi 4

$$f(x) = \frac{|x+x^2|}{|x-x^2|}$$

```
>$showev('limit((abs(x+x^2)) / (abs(x-x^2)), x, 0))
```

$$\lim_{x \rightarrow 0} \frac{|x^2 + x|}{|x^2 - x|} = 1$$

```
>plot2d("((x+x^2) / (x-x^2))", -10, 20)
```

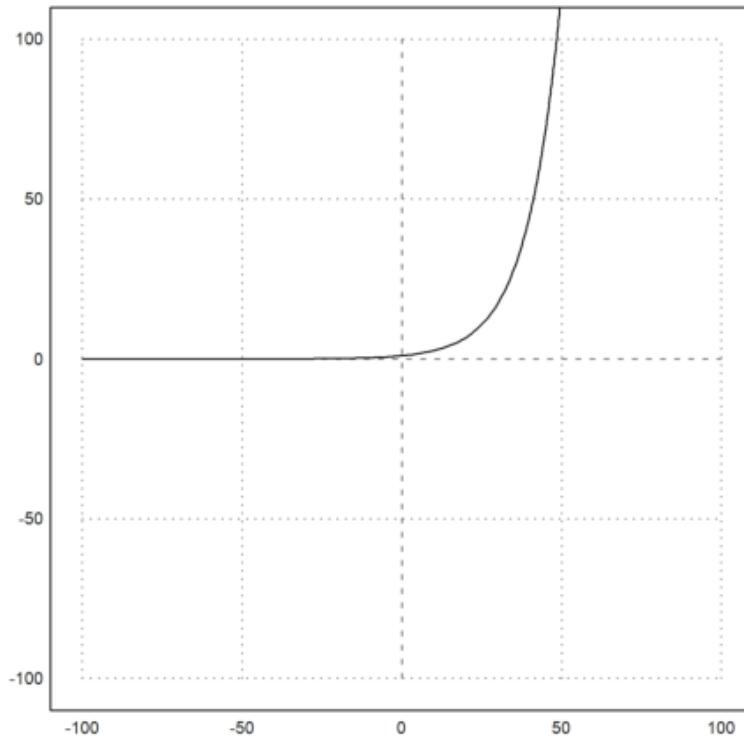
Fungsi 5

$$f(x) = \frac{1}{\sin^x 2}$$

```
>$showev('limit(sin(2)^(-x), x, 0))
```

$$\lim_{x \rightarrow 0} \frac{1}{\sin^x 2} = 1$$

```
>plot2d("(sin(2)^(-x))", -100, 100, -100, 100):
```



Turunan Fungsi

Definisi turunan:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

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
```

$$\lim_{h \rightarrow 0} \frac{(x+h)^2 - x^2}{h} = 2x$$

```
>p &= expand((x+h)^2-x^2)|simplify; $p // pembilang dijabarkan dan disederhanakan
```

$$2hx + h^2$$

```
>q &=ratsimp(p/h); $q // ekspresi yang akan dihitung limitnya disederhanakan
```

$$2x + h$$

```
>$limit(q,h,0) // nilai limit sebagai turunan
```

$2x$

```
>$showev('limit(((x+h)^n-x^n)/h,h,0)) // turunan x^n
```

$$\lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h} = n x^{n-1}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, ekspansikan $(x+h)^n$ dengan menggunakan teorema binomial.

```
>$showev('limit((sin(x+h)-sin(x))/h,h,0)) // turunan sin(x)
```

$$\lim_{h \rightarrow 0} \frac{\sin(x+h) - \sin x}{h} = \cos x$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini. Sebagai petunjuk, ekspansikan $\sin(x+h)$ dengan menggunakan rumus jumlah dua sudut.

```
>$showev('limit((log(x+h)-log(x))/h,h,0)) // turunan log(x)
```

$$\lim_{h \rightarrow 0} \frac{\log(x+h) - \log x}{h} = \frac{1}{x}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, gunakan sifat-sifat logaritma dan hasil limit pada bagian sebelumnya di atas.

```
>$showev('limit((1/(x+h)-1/x)/h,h,0)) // turunan 1/x
```

$$\lim_{h \rightarrow 0} \frac{\frac{1}{x+h} - \frac{1}{x}}{h} = -\frac{1}{x^2}$$

```
>$showev('limit((E^(x+h)-E^x)/h,h,0)) // turunan f(x)=e^x
```

Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"
Answering "Is x an integer?" with "integer"

```

Answering "Is x an integer?" with "integer"
Maxima is asking
Acceptable answers are: yes, y, Y, no, n, N, unknown, uk
Is x an integer?

```

Use assume!
Error in:
\$showev('limit((E^(x+h)-E^x)/h,h,0)) // turunan f(x)=e^x ...
^

Maxima bermasalah dengan limit:

$$\lim_{h \rightarrow 0} \frac{e^{x+h} - e^x}{h}.$$

Oleh karena itu diperlukan trik khusus agar hasilnya benar.

```
>$showev('limit((E^h-1)/h,h,0))
```

$$\lim_{h \rightarrow 0} \frac{e^h - 1}{h} = 1$$

```
>$showev('factor(E^(x+h)-E^x))
```

$$factor(e^{x+h} - e^x) = (e^h - 1) e^x$$

```
>$showev('limit(factor((E^(x+h)-E^x)/h),h,0)) // turunan f(x)=e^x
```

$$\left(\lim_{h \rightarrow 0} \frac{e^h - 1}{h} \right) e^x = e^x$$

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

$$\begin{matrix} x \\ x \end{matrix}$$

```
>$showev('limit(f(x),x,0))
```

$$\lim_{x \rightarrow 0} x^x = 1$$

Silakan Anda gambar kurva

$$y = x^x.$$

```
>$showev('limit((f(x+h)-f(x))/h,h,0)) // turunan f(x)=x^x
```

$$\lim_{h \rightarrow 0} \frac{(x + h)^{x+h} - x^x}{h} = infinity$$

Di sini Maxima juga bermasalah terkait limit:

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h}.$$

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
```

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h} = x^x (\log x + 1)$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

```
>&forget(x>0) // jangan lupa, lupakan asumsi untuk kembali ke semula
```

[x > 0]

```
>&forget(x<0)
```

[x < 0]

```
>&facts()
```

[]

```
>$showev('limit((asin(x+h)-asin(x))/h,h,0)) // turunan arcsin(x)
```

$$\lim_{h \rightarrow 0} \frac{\arcsin(x+h) - \arcsin x}{h} = \frac{1}{\sqrt{1-x^2}}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

```
>$showev('limit((tan(x+h)-tan(x))/h,h,0)) // turunan tan(x)
```

$$\lim_{h \rightarrow 0} \frac{\tan(x+h) - \tan x}{h} = \frac{1}{\cos^2 x}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

```
>function f(x) &= sinh(x) // definisikan f(x)=sinh(x)
```

$$\sinh(x)$$

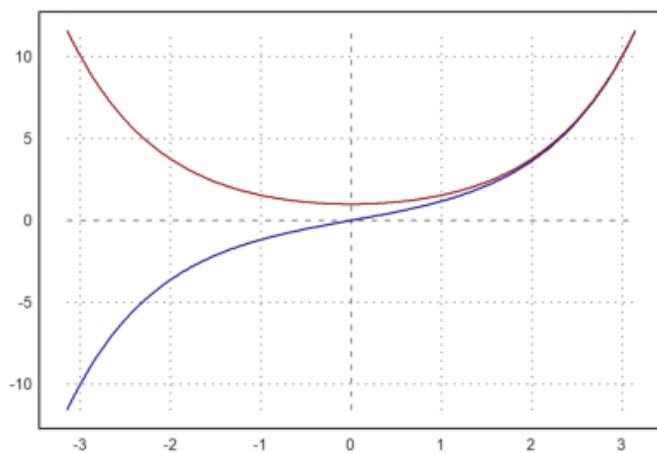
```
>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x) // df(x) = f'(x)
```

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

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
```

$$\sin^2(3x^5 + 7)$$

```
>diff(f,3), diffc(f,3)
```

1198.32948904
1198.72863721

Apakah perbedaan diff dan diffc?

```
>\$showev('diff(f(x),x))
```

$$\frac{d}{dx} \sin^2(3x^5 + 7) = 30x^4 \cos(3x^5 + 7) \sin(3x^5 + 7)$$

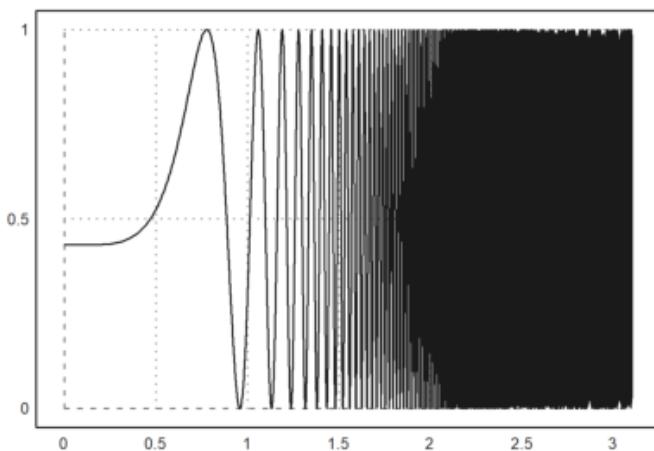
```
>\$% with x=3
```

$$\%at \left(\frac{d}{dx} \sin^2(3x^5 + 7), x = 3 \right) = 2430 \cos 736 \sin 736$$

```
>\$float(%)
```

$$\%at \left(\frac{d^{1.0}}{dx^{1.0}} \sin^2(3.0x^5 + 7.0), x = 3.0 \right) = 1198.728637211748$$

```
>plot2d(f,0,3.1):
```



```
>function f(x) &=5*cos(2*x)-2*x*sin(2*x) // mendefinisikan fungsi f
```

$$5 \cos(2x) - 2x \sin(2x)$$

```
>function df(x) &=diff(f(x),x) // fd(x) = f'(x)
```

$$- 12 \sin(2x) - 4x \cos(2x)$$

```
>${'f(1)=f(1), $float(f(1)), '$f(2)=f(2), $float(f(2)) // nilai f(1) dan f(2)}
```

-0.2410081230863468

images/Shalih Abdillah_22305144009_ApKom5-101-lar

images/Shalih Abdillah_22305144009_ApKom5-102-lar

images/Shalih Abdillah_22305144009_ApKom5-103-lar

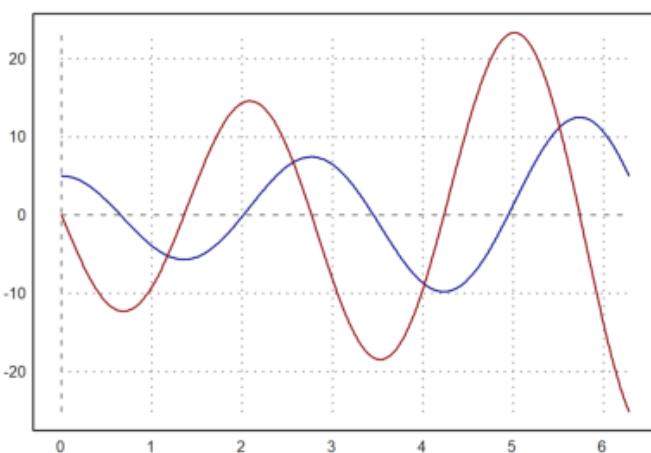
```
>xp=solve("df(x)",1,2,0) // solusi f'(x)=0 pada interval [1, 2]
```

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
```



Perhatikan titik-titik "puncak" grafik $y=f(x)$ dan nilai turunan pada saat grafik fungsinya mencapai titik "puncak" tersebut.

```
>
```

Latihan

Bukalah buku Kalkulus. Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi). Untuk setiap fungsi, tentukan turunannya dengan menggunakan definisi turunan (limit), menggunakan perintah diff, dan secara manual (langkah demi langkah yang dihitung dengan Maxima) seperti contoh-contoh di atas. Gambar grafik fungsi asli dan fungsi turunannya pada sumbu koordinat yang sama.

Fungsi 1

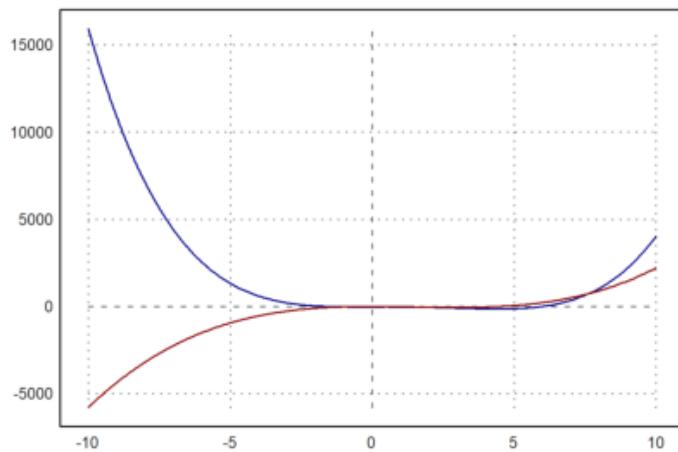
```
>function f(x) &= (x^3+4)*(x-6); $f(x)
```

$$(x - 6) (x^3 + 4)$$

```
>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x)
```

$$4x^3 - 18x^2 + 4$$

```
>aspect(1.5); plot2d(["f(x)", "df(x)"], -10, 10, color=[blue, red]):
```



Fungsi 2

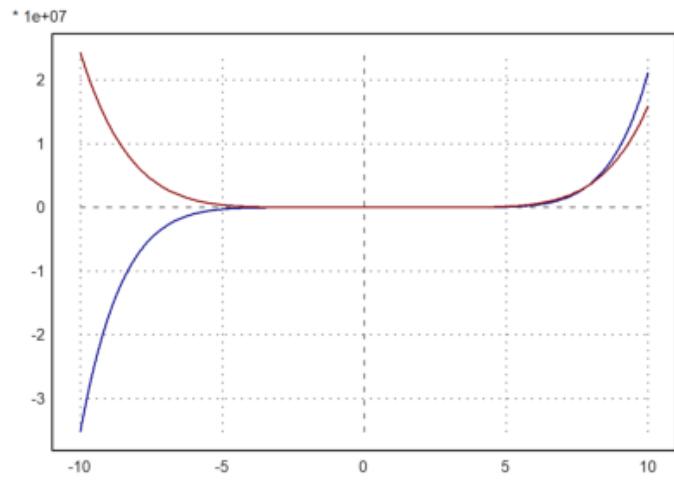
```
>function f(x) &= (3*x^3+2*x^2-x)*(x^4-3*x^3-4*x^2+1); $f(x)
```

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

```
>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x)
```

$$21x^6 - 42x^5 - 95x^4 - 20x^3 + 21x^2 + 4x - 1$$

```
>aspect(1.5); plot2d(["f(x)", "df(x)"], -10, 10, color=[blue, red]):
```



Fungsi 3

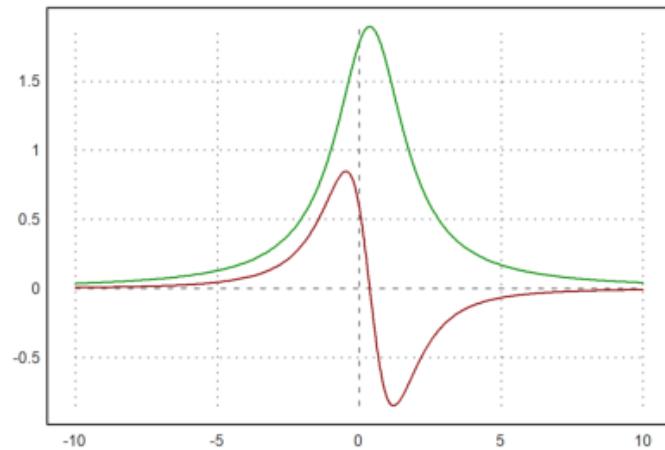
```
>function f(x) &= (16)/(4*x^2-3*x+9); $f(x)
```

$$\frac{16}{4x^2 - 3x + 9}$$

```
>function df(x) &= limit((f(x+h)-f(x))/h, h, 0); $df(x)
```

$$\frac{48 - 128x}{16x^4 - 24x^3 + 81x^2 - 54x + 81}$$

```
>aspect(1.5); plot2d(["f(x)", "df(x)"], -10, 10, color=[green, red]):
```



Fungsi 4

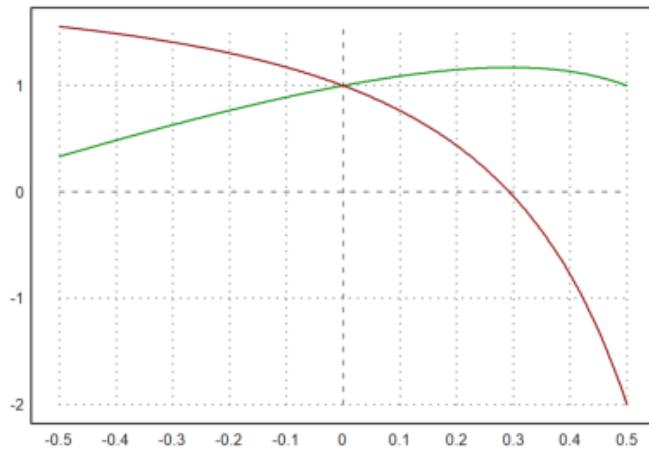
```
>function f(x) &= (2*x^2-1)/(x-1); $f(x)
```

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

```
>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x)
```

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

```
>aspect(1.5); plot2d(["f(x)", "df(x)"], -0.5, 0.5, color=[green, red]):
```



Fungsi 5

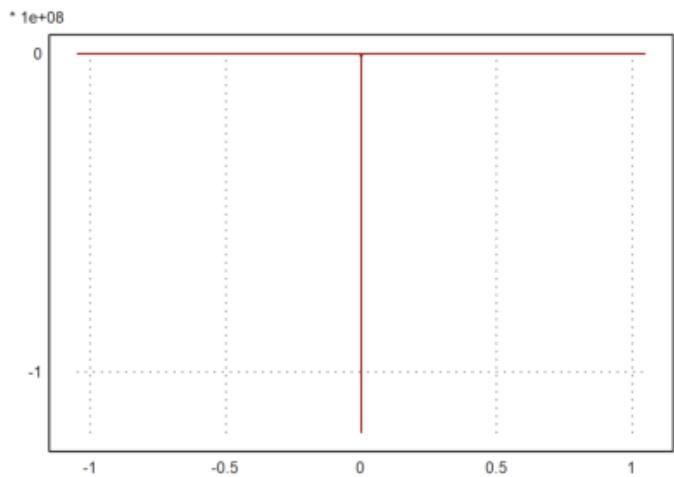
```
>function f(x) &= (sin(x)+cos(x))/(sin(x)); $f(x)
```

$$\frac{\sin x + \cos x}{\sin x}$$

```
>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x)
```

$$\frac{-\sin^2 x - \cos^2 x}{\sin^2 x}$$

```
>aspect(1.5); plot2d(["f(x)", "df(x)"], -pi/3, pi/3, color=[green, red]):
```



>

Integral

EMT dapat digunakan untuk menghitung integral, baik integral tak tentu maupun integral tentu. Untuk integral tak tentu (simbolik) sudah tentu EMT menggunakan Maxima, sedangkan untuk perhitungan integral tentu EMT sudah menyediakan beberapa fungsi yang mengimplementasikan algoritma kuadratur (perhitungan integral tentu menggunakan metode numerik).

Pada notebook ini akan ditunjukkan perhitungan integral tentu dengan menggunakan Teorema Dasar Kalkulus:

$$\int_a^b f(x) dx = F(b) - F(a), \quad \text{dengan } F'(x) = f(x).$$

Fungsi untuk menentukan integral adalah `integrate`. Fungsi ini dapat digunakan untuk menentukan, baik integral tentu maupun tak tentu (jika fungsinya memiliki antiderivatif). Untuk perhitungan integral tentu fungsi `integrate` menggunakan metode numerik (kecuali fungsinya tidak integrabel, kita tidak akan menggunakan metode ini).

```
>$showev('integrate(x^n,x))
```

Answering "Is n equal to -1?" with "no"

$$\int x^n dx = \frac{x^{n+1}}{n+1}$$

```
>$showev('integrate(1/(1+x),x))
```

$$\int \frac{1}{x+1} dx = \log(x+1)$$

```
>$showev('integrate(1/(1+x^2),x))
```

$$\int \frac{1}{x^2 + 1} dx = \arctan x$$

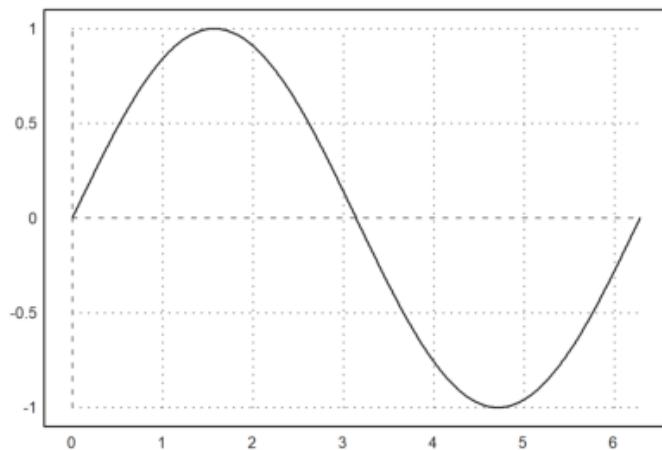
```
>showev('integrate(1/sqrt(1-x^2),x))
```

$$\int \frac{1}{\sqrt{1-x^2}} dx = \arcsin x$$

```
>showev('integrate(sin(x),x,0,pi))
```

$$\int_0^\pi \sin x dx = 2$$

```
>plot2d("sin(x)",0,2*pi):
```



```
>showev('integrate(sin(x),x,a,b))
```

$$\int_a^b \sin x dx = \cos a - \cos b$$

```
>showev('integrate(x^n,x,a,b))
```

Answering "Is n positive, negative or zero?" with "positive"

$$\int_a^b x^n dx = \frac{b^{n+1}}{n+1} - \frac{a^{n+1}}{n+1}$$

```
>showev('integrate(x^2*sqrt(2*x+1),x))
```

$$\int x^2 \sqrt{2x+1} dx = \frac{(2x+1)^{\frac{7}{2}}}{28} - \frac{(2x+1)^{\frac{5}{2}}}{10} + \frac{(2x+1)^{\frac{3}{2}}}{12}$$

```
>\$showev('integrate(x^2*sqrt(2*x+1),x,0,2))
```

$$\int_0^2 x^2 \sqrt{2x+1} dx = \frac{25^{\frac{5}{2}}}{21} - \frac{2}{105}$$

```
>\$ratsimp(%)
```

$$\int_0^2 x^2 \sqrt{2x+1} dx = \frac{25^{\frac{7}{2}} - 2}{105}$$

```
>\$showev('integrate((sin(sqrt(x)+a)*E^sqrt(x))/sqrt(x),x,0,pi^2))
```

$$\int_0^{\pi^2} \frac{\sin(\sqrt{x}+a) e^{\sqrt{x}}}{\sqrt{x}} dx = (-e^\pi - 1) \sin a + (e^\pi + 1) \cos a$$

```
>\$factor(%)
```

$$\int_0^{\pi^2} \frac{\sin(\sqrt{x}+a) e^{\sqrt{x}}}{\sqrt{x}} dx = (-e^\pi - 1) (\sin a - \cos a)$$

```
>function map f(x) &= E^(-x^2)
```

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

```
>\$showev('integrate(f(x),x))
```

$$\int e^{-x^2} dx = \frac{\sqrt{\pi} \operatorname{erf}(x)}{2}$$

Fungsi f tidak memiliki antiturunan, integralnya masih memuat integral lain.

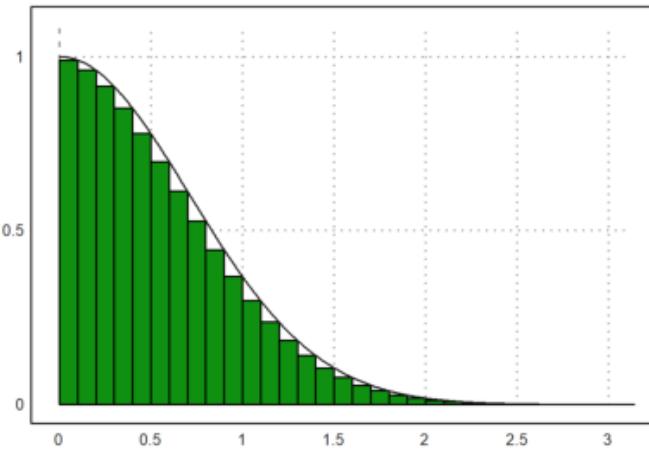
$$\operatorname{erf}(x) = \int \frac{e^{-x^2}}{\sqrt{\pi}} dx.$$

Kita tidak dapat menggunakan teorema Dasar kalkulus untuk menghitung integral tentu fungsi tersebut jika semua batasnya berhingga. Dalam hal ini dapat digunakan metode numerik (rumus kuadratur).

Misalkan kita akan menghitung:

$$\int_0^{\pi} \frac{\sin x + \cos x}{\sin x} dx$$

```
>x=0:0.1:pi-0.1; plot2d(x,f(x+0.1),>bar); plot2d("f(x)",0,pi,>add):
```



Integral tentu

$$\int_0^\pi \frac{\sin x + \cos x}{\sin x} dx$$

dapat dihampiri dengan jumlah luas persegi-persegi panjang di bawah kurva $y=f(x)$ tersebut. Langkah-langkahnya adalah sebagai berikut.

```
>t &= makelist(a,a,0,pi-0.1,0.1); // t sebagai list untuk menyimpan nilai-nilai x
>fx &= makelist(f(t[i]+0.1),i,1,length(t)); // simpan nilai-nilai f(x)
>/> jangan menggunakan x sebagai list, kecuali Anda pakar Maxima!
```

Hasilnya adalah:

$$\int_0^\pi \frac{\sin x + \cos x}{\sin x} dx = 1.22817137704692$$

Jumlah tersebut diperoleh dari hasil kali lebar sub-subinterval (=0.1) dan jumlah nilai-nilai $f(x)$ untuk $x = 0.1, 0.2, 0.3, \dots, 3.2$.

```
>0.1*sum(f(x+0.1)) // cek langsung dengan perhitungan numerik EMT
```

0.836219610253

Untuk mendapatkan nilai integral tentu yang mendekati nilai sebenarnya, lebar sub-intervalnya dapat diperkecil lagi, sehingga daerah di bawah kurva tertutup semuanya, misalnya dapat digunakan lebar subinterval 0.001. (Silakan dicoba!)

Meskipun Maxima tidak dapat menghitung integral tentu fungsi tersebut untuk batas-batas yang berhingga, namun integral tersebut dapat dihitung secara eksak jika batas-batasnya tak hingga. Ini adalah salah satu keajaiban di dalam matematika, yang terbatas tidak dapat dihitung secara eksak, namun yang tak hingga malah dapat dihitung secara eksak.

```
>$showev('integrate(f(x),x,0,inf))
```

$$\int_0^\infty e^{-x^2} dx = \frac{\sqrt{\pi}}{2}$$

Tunjukkan kebenaran hasil di atas!

Berikut adalah contoh lain fungsi yang tidak memiliki antiderivatif, sehingga integral tentunya hanya dapat dihitung dengan metode numerik.

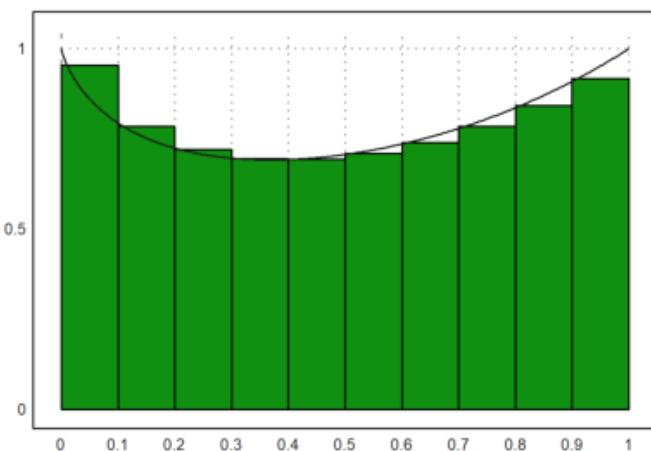
```
>function f(x) &= x^x
```

$$\frac{x}{x}$$

```
>$showev('integrate(f(x),x,0,1))
```

$$\int_0^1 x^x \, dx = \int_0^1 x^x \, dx$$

```
>x=0:0.1:1-0.01; plot2d(x,f(x+0.01),>bar); plot2d("f(x)",0,1,>add):
```



Maxima gagal menghitung integral tentu tersebut secara langsung menggunakan perintah integrate. Berikut kita lakukan seperti contoh sebelumnya untuk mendapat hasil atau pendekatan nilai integral tentu tersebut.

```
>t &= makelist(a,a,0,1-0.01,0.01);
>fx &= makelist(f(t[i]+0.01),i,1,length(t));
```

$$\int_0^1 \frac{\sin x + \cos x}{\sin x} \, dx = 0.122817137704692$$

Apakah hasil tersebut cukup baik? perhatikan gambarnya.

```
>function f(x) &= sin(3*x^5+7)^2
```

$$\sin^2(x^5 + 7)$$

```
>integrate(f,0,1)
```

0.542581176074

```
>&showev('integrate(f(x),x,0,1))
```

$$\begin{aligned} & \int_0^{1/5} \frac{\sin^2(3x^5 + 7)}{10^6} dx = \frac{1}{10} \operatorname{gamma}(-, 5) \sin(14) \sin(\pi) \\ & - \frac{((6 \operatorname{gamma_incomplete}(-, 6I) + 6 \operatorname{gamma_incomplete}(-, -6I)) \\ & \quad \sin(14) + (6 \operatorname{gamma_incomplete}(-, 6I) \\ & \quad - 6 \operatorname{gamma_incomplete}(-, -6I)) \cos(14)) \sin(\pi) - 60)/120}{5^4} \\ & \quad 10^4 \end{aligned}$$

```
>&float(%)
```

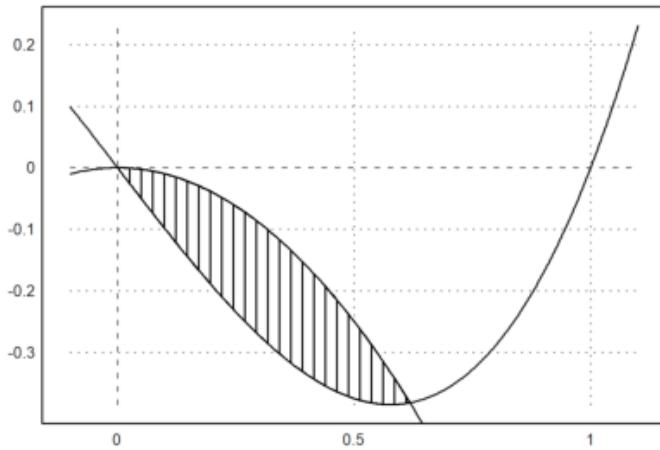
$$\begin{aligned} & \int_0^{1.0} \frac{\sin^2(3.0x^5 + 7.0)}{10^6} dx = \\ & 0.09820784258795788 - 0.00833333333333333 \\ & (0.3090169943749474 (0.1367372182078336 \\ & (4.192962712629476 I \operatorname{gamma_incomplete}(0.2, 6.0I) \\ & - 4.192962712629476 I \operatorname{gamma_incomplete}(0.2, -6.0I)) \\ & + 0.9906073556948704 (4.192962712629476 \operatorname{gamma_incomplete}(0.2, 6.0I) \\ & + 4.192962712629476 \operatorname{gamma_incomplete}(0.2, -6.0I))) - 60.0) \end{aligned}$$

```
>$showev('integrate(x*exp(-x),x,0,1)) // Integral tentu (eksak)
```

$$\int_0^1 x e^{-x} dx = 1 - 2e^{-1}$$

Aplikasi Integral Tentu

```
>plot2d("x^3-x", -0.1, 1.1); plot2d("-x^2", >add); ...
>b=solve("x^3-x+x^2", 0.5); x=linspace(0,b,200); xi=flipx(x); ...
>plot2d(x|xi, x^3-x|-xi^2, >filled, style="|", fillcolor=1, >add); // Plot daerah antara 2 kurva
```



```
>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
```

Hasil tersebut akan kita bandingkan dengan perhitungan secara analitik.

```
>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(%)}
```

$$\int_{0.0}^{0.6180339887498949} -1.0 x^3 - 1.0 x^2 + x \, dx = 0.07581917135421037$$

Panjang Kurva

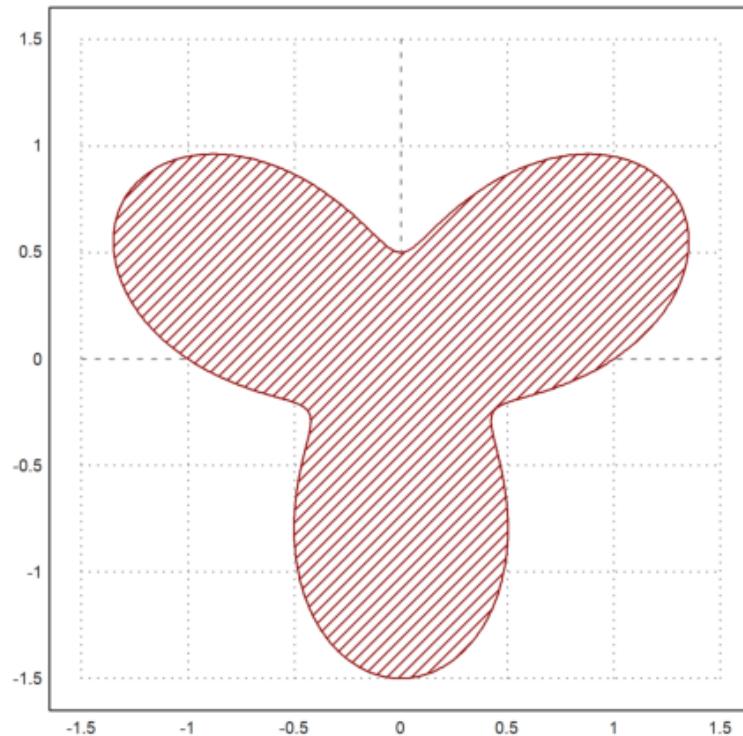
Hitunglah panjang kurva berikut ini dan luas daerah di dalam kurva tersebut.

$$\gamma(t) = (r(t) \cos(t), r(t) \sin(t))$$

dengan

$$r(t) = 1 + \frac{\sin(3t)}{2}, \quad 0 \leq t \leq 2\pi.$$

```
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t); ...
>plot2d(x,y,>filled,fillcolor=red,style="/"',r=1.5): // Kita gambar kurvanya terlebih dahulu
```



```
>function r(t) &= 1+sin(3*t)/2; $'r(t)=r(t)
```

$r([0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.7999999999999999, 0.8999999999999999, 0.9999999999999999, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7,$

```
>function fx(t) &= r(t)*cos(t); $' fx(t)=fx(t)
```

$fx([0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.7999999999999999, 0.8999999999999999, 0.9999999999999999, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7])$

```
>function fy(t) &= r(t)*sin(t); $' fy(t)=fy(t)
```

$fy([0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.7999999999999999, 0.8999999999999999, 0.9999999999999999, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7])$

```
>function ds(t) &= trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); $' ds(t)=ds(t)
```

Maxima said:

```
diff: second argument must be a variable; found  
[0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.7999999999999999,  
 0.8999999999999999,0.9999999999999999,1.1,1.2,1.3,1.4,1.5,1.6,  
 1.7,1.8,1.900000000000001,2.0,2.100000000000001,  
 2.200000000000001,2.300000000000001,2.400000000000001,  
 2.500000000000001,2.600000000000001,2.700000000000001,  
 2.800000000000001,2.900000000000001,3.000000000000001]  
-- an error. To debug this try: debugmode(true);
```

Error in:

```
... e(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); $' ds(t)=ds(t ...  
^
```

```
>$integrate(ds(x),x,0,2*pi) //panjang (keliling) kurva
```

Maxima said:

```
defint: variable of integration must be a simple or subscripted variable.  
defint: found errexpl  
-- an error. To debug this try: debugmode(true);
```

Error in:

```
$integrate(ds(x),x,0,2*pi) //panjang (keliling) kurva ...  
^
```

Maxima gagal melakukan perhitungan eksak integral tersebut.

Berikut kita hitung integralnya secara umerik dengan perintah EMT.

```
>integrate("ds(x)",0,2*pi)
```

Function ds not found.

Try list ... to find functions!

Error in expression: ds(x)

```
%mapexpression1:
```

```
    return expr(x,args());
```

Error in map.

```
%evalexpression:
```

```

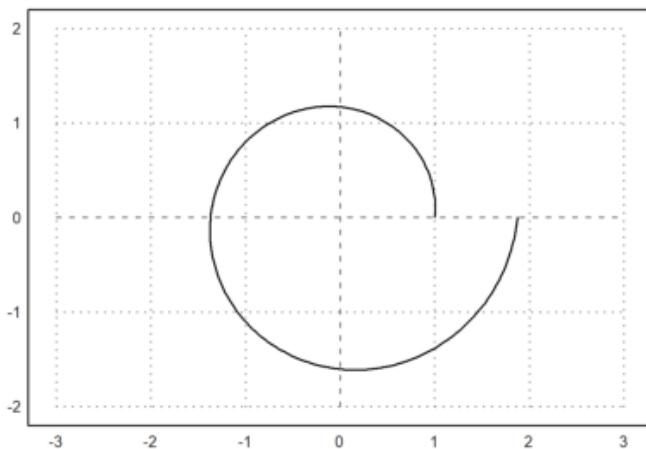
if maps then return %mapexpression1(x,f$;args());
gauss:
if maps then y=%evalexpression(f$,a+h-(h*xn)',maps;args());
adaptivegauss:
t1=gauss(f$,c,c+h;args(),=maps);
Try "trace errors" to inspect local variables after errors.
integrate:
return adaptivegauss(f$,a,b,eps*1000;args(),=maps);

```

Spiral Logaritmik

$$x = e^{ax} \cos x, y = e^{ax} \sin x.$$

```
>a=0.1; plot2d("exp(a*x)*cos(x)", "exp(a*x)*sin(x)", r=2, xmin=0, xmax=2*pi):
```



```
>&kill(a) // hapus expresi a
```

done

```
>function fx(t) &= exp(a*t)*cos(t); $'fx(t)=fx(t)
```

```
fx ([0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.7999999999999999,0.8999999999999999,0.9999999999999999,1.1,1.2,1.3,1.4,1.5,1.6,1.7]
```

```
>function fy(t) &= exp(a*t)*sin(t); $'fy(t)=fy(t)
```

```
fy ([0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.7999999999999999,0.8999999999999999,0.9999999999999999,1.1,1.2,1.3,1.4,1.5,1.6,1.7]
```

```
>function df(t) &= trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); $'df(t)=df(t)
```

```

Maxima said:
diff: second argument must be a variable; found
[0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.7999999999999999,
 0.8999999999999999,0.9999999999999999,1.1,1.2,1.3,1.4,1.5,1.6,
 1.7,1.8,1.9000000000000001,2.0,2.1000000000000001,
 2.2000000000000001,2.3000000000000001,2.4000000000000001,
 2.5000000000000001,2.6000000000000001,2.7000000000000001,
 2.8000000000000001,2.9000000000000001,3.0000000000000001]
-- an error. To debug this try: debugmode(true);

Error in:
... e(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); $'df(t)=df(t ...
^

```

```
>S &=integrate(df(t),t,0,2*pi); $S // panjang kurva (spiral)
```

```

Maxima said:
expt: undefined: 0 to a negative exponent.
#0: df(x=[0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.7999999999999999,0.8999999999999999,0.9999999999999999,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9000000000000001,2.0,2.1000000000000001,2.2000000000000001,2.3000000000000001,2.4000000000000001,2.5000000000000001,2.6000000000000001,2.7000000000000001,2.8000000000000001,2.9000000000000001,3.0000000000000001])
-- an error. To debug this try: debugmode(true);

Error in:
S &=integrate(df(t),t,0,2*pi); $S // panjang kurva (spiral) ...
^

```

```
>S(a=0.1) // Panjang kurva untuk a=0.1
```

```

Function S not found.
Try list ... to find functions!
Error in:
S(a=0.1) // Panjang kurva untuk a=0.1 ...
^

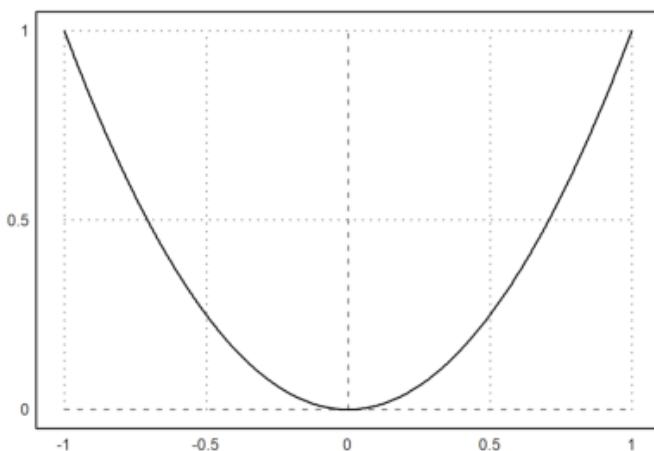
```

Soal:

Tunjukkan bahwa keliling lingkaran dengan jari-jari r adalah $K=2\pi r$.

Berikut adalah contoh menghitung panjang parabola.

```
>plot2d("x^2",xmin=-1,xmax=1):
```



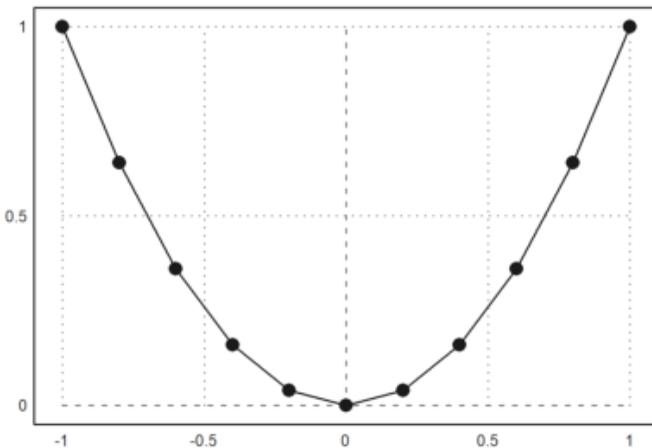
```
>$showev('integrate(sqrt(1+diff(x^2,x)^2),x,-1,1))
```

$$\int_{-1}^1 \sqrt{4x^2 + 1} dx = \frac{\operatorname{asinh} 2 + 2\sqrt{5}}{2}$$

```
>$float(%)
```

$$\int_{-1.0}^{1.0} \sqrt{4.0x^2 + 1.0} dx = 2.957885715089195$$

```
>x=-1:0.2:1; y=x^2; plot2d(x,y); ...
> plot2d(x,y,points=1,style="o#",add=1):
```



Panjang tersebut dapat dihampiri dengan menggunakan jumlah panjang ruas-ruas garis yang menghubungkan titik-titik pada parabola tersebut.

```
>i=1:cols(x)-1; sum(sqrt((x[i+1]-x[i])^2+(y[i+1]-y[i])^2))
```

2.95191957027

Hasilnya mendekati panjang yang dihitung secara eksak. Untuk mendapatkan hampiran yang cukup akurat, jarak antar titik dapat diperkecil, misalnya 0.1, 0.05, 0.01, dan seterusnya. Cobalah Anda ulangi perhitungannya dengan nilai-nilai tersebut.

Koordinat Kartesius

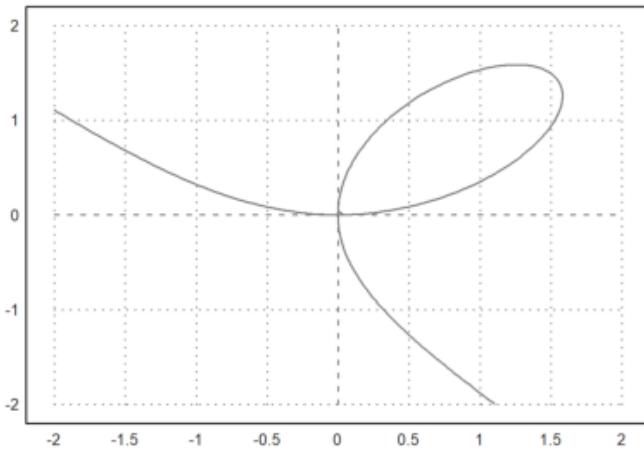
Berikut diberikan contoh perhitungan panjang kurva menggunakan koordinat Kartesius. Kita akan hitung panjang kurva dengan persamaan implisit:

$$x^3 + y^3 - 3xy = 0.$$

```
>z &= x^3+y^3-3*x*y; $z
```

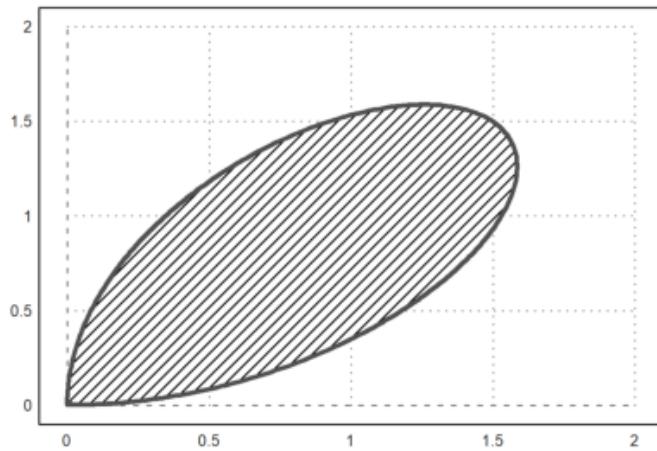
$$y^3 - 3xy + x^3$$

```
>plot2d(z, r=2, level=0, n=100):
```



Kita tertarik pada kurva di kuadran pertama.

```
>plot2d(z, a=0, b=2, c=0, d=2, level=[-10; 0], n=100, contourwidth=3, style="/" ):
```



Kita selesaikan persamaannya untuk x.

```
>$z with y=l*x, sol &= solve(%,x); $sol
```

$$\left[x = \frac{3l}{l^3 + 1}, x = 0 \right]$$

[images/Shalih Abdillah_22305144009_ApKom5-166-large.png](#)

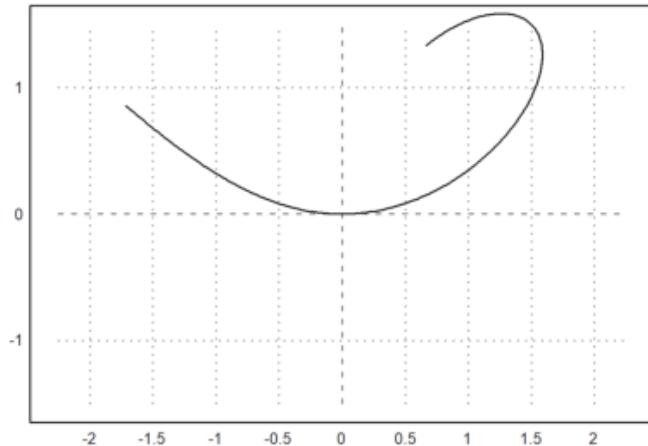
Kita gunakan solusi tersebut untuk mendefinisikan fungsi dengan Maxima.

```
>function f(l) &= rhs(sol[1]); $'f(l)=f(l)
```

$$f(l) = \frac{3l}{l^3 + 1}$$

Fungsi tersebut juga dapat digunakan untuk menggambar kurvanya. Ingat, bahwa fungsi tersebut adalah nilai x dan nilai $y=l^*x$, yakni $x=f(l)$ dan $y=l^*f(l)$.

```
>plot2d(&f(x), &x*f(x), xmin=-0.5, xmax=2, a=0, b=2, c=0, d=2, r=1.5);
```



Elemen panjang kurva adalah:

$$ds = \sqrt{f'(l)^2 + (lf'(l) + f(l))^2}.$$

```
>function ds(l) &= ratsimp(sqrt(diff(f(l), l)^2 + diff(l*f(l), l)^2)); $'ds(l)=ds(l)
```

$$ds(l) = \frac{\sqrt{9l^8 + 36l^6 - 36l^5 - 36l^3 + 36l^2 + 9}}{\sqrt{l^{12} + 4l^9 + 6l^6 + 4l^3 + 1}}$$

```
>$integrate(ds(l), l, 0, 1)
```

$$\int_0^1 \frac{\sqrt{9l^8 + 36l^6 - 36l^5 - 36l^3 + 36l^2 + 9}}{\sqrt{l^{12} + 4l^9 + 6l^6 + 4l^3 + 1}} dl$$

Integral tersebut tidak dapat dihitung secara eksak menggunakan Maxima. Kita hitung integral tersebut 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
```

4.91748872168

Perhitungan di datas dapat dilakukan untuk sebarang fungsi x dan y dengan mendefinisikan fungsi EMT, misalnya kita beri nama panjangkurva. Fungsi ini selalu memanggil Maxima untuk menurunkan fungsi yang diberikan.

```
>function panjangkurva(fx, fy, a, b) ...
```

```
ds=mxm("sqrt(diff(@fx,x)^2+diff(@fy,x)^2)");
return romberg(ds,a,b);
endfunction
```

```
>panjangkurva("x", "x^2", -1, 1) // cek untuk menghitung panjang kurva parabola sebelumnya
```

2.95788571509

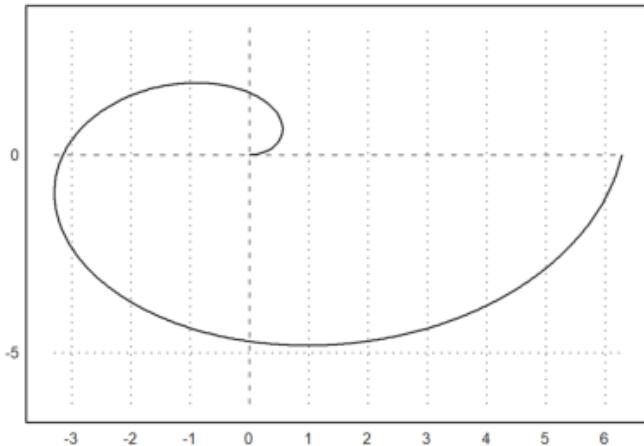
Bandingkan dengan nilai eksak di atas.

```
>2*panjangkurva(mxm("f(x)", mxm("x*f(x)", 0, 1)) // cek contoh terakhir, bandingkan hasilnya
```

4.91748872168

Kita hitung panjang spiral Archimedes berikut ini dengan fungsi tersebut.

```
>plot2d("x*cos(x)", "x*sin(x)", xmin=0, xmax=2*pi, square=1):
```



```
>panjangkurva("x*cos(x)", "x*sin(x)", 0, 2*pi)
```

21.2562941482

Berikut kita definisikan fungsi yang sama namun dengan Maxima, untuk perhitungan eksak.

```
>&kill(ds,x,fx,fy)
```

done

```
>function ds(fx,fy) &=& sqrt(diff(fx,x)^2+diff(fy,x)^2)
```

$$\sqrt{(\cos x - x \sin x)^2 + (\sin x + x \cos x)^2}$$

```
>sol &= ds(x*cos(x),x*sin(x)); $sol // Kita gunakan untuk menghitung panjang kurva terakhir
```

$$\sqrt{(\cos x - x \sin x)^2 + (\sin x + x \cos x)^2}$$

```
>$sol | trigreduce | expand, $integrate(% ,x,0,2*pi), %()
```

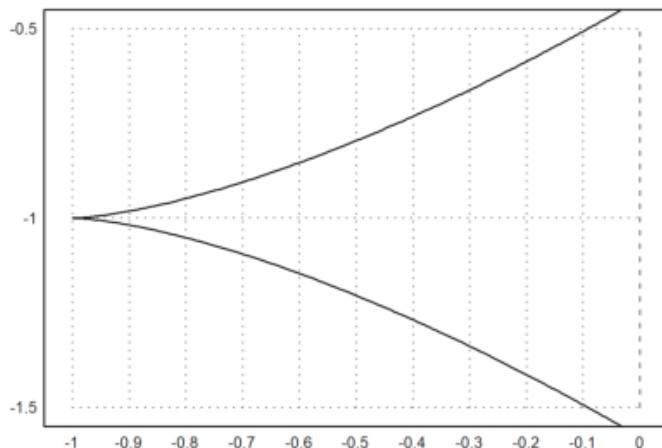
$$\frac{\operatorname{asinh}(2\pi) + 2\pi\sqrt{4\pi^2 + 1}}{2}$$

[images/Shalih Abdillah_22305144009_ApKom5-174-large.png](#)

21.2562941482

Hasilnya sama dengan perhitungan menggunakan fungsi EMT.
Berikut adalah contoh lain penggunaan fungsi Maxima tersebut.

```
>plot2d("3*x^2-1","3*x^3-1",xmin=-1/sqrt(3),xmax=1/sqrt(3),square=1):
```



```
>sol &= radcan(ds(3*x^2-1, 3*x^3-1)); $sol
```

$$3x\sqrt{9x^2 + 4}$$

```
>$showev('integrate(sol,x,0,1/sqrt(3))), $2*float(%); // panjang kurva di atas
```

$$6.0 \int_{0.0}^{0.5773502691896258} x \sqrt{9.0x^2 + 4.0} dx = 2.337835372767141$$

[images/Shalih Abdillah_22305144009_ApKom5-178-larg]

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, 1.58850117255346 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]

```

>1

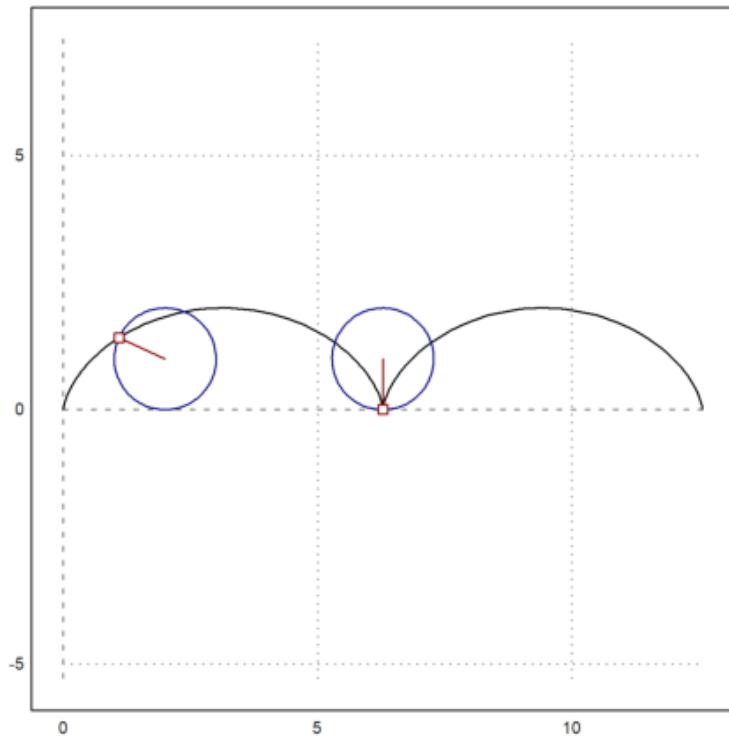
1

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):

```



Berikut dihitung panjang lintasan untuk 1 putaran penuh. (Jangan salah menduga bahwa panjang lintasan 1 putaran penuh sama dengan keliling lingkaran!)

```

>ds &= radcan(sqrt(diff(ex,x)^2+diff(ey,x)^2)); $ds=trigsimp(ds) // elemen panjang kurva s

```

$$\sqrt{\sin^2 x + \cos^2 x - 2 \cos x + 1} = \sqrt{2 - 2 \cos x}$$

```
>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

Perhatikan, seperti terlihat pada gambar, panjang sikloid lebih besar daripada keliling lingkarannya, yakni: 2π .

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 2π sejauh $2\pi R$.)

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 $\mathbf{T}'(s)$ ada. Vektor ini merupakan normal kurva yang arahnya menuju pusat kurvatur, norm-nya merupakan nilai kurvatur (kelengkungan):

$$\begin{aligned}\mathbf{T}(s) &= \gamma'(s), \\ \mathbf{T}^2(s) &= 1 \text{ (konstanta)} \Rightarrow \mathbf{T}'(s) \cdot \mathbf{T}(s) = 0 \\ \kappa(s) &= \|\mathbf{T}'(s)\| = \|\gamma''(s)\| = \sqrt{x''(s)^2 + y''(s)^2}.\end{aligned}$$

Nilai

$$R(s) = \frac{1}{\kappa(s)}$$

disebut jari-jari (radius) kelengkungan kurva.

Bilangan riil

$$k(s) = \pm \kappa(s)$$

disebut nilai kelengkungan bertanda.

Contoh:

Akan ditentukan kurvatur lingkaran

$$x = r \cos t, y = r \sin t.$$

```
>fx &= r*cos(t); fy &=r*sin(t);
>&assume(t>0,r>0); s &=integrate(sqrt(diff(fx,t)^2+diff(fy,t)^2),t,0,t); s // elemen panjang
```

Maxima said:

```
diff: second argument must be a variable; found
[0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.7999999999999999,
 0.8999999999999999,0.9999999999999999,1.1,1.2,1.3,1.4,1.5,1.6,
 1.7,1.8,1.9000000000000001,2.0,2.1000000000000001,
 2.2000000000000001,2.3000000000000001,2.4000000000000001,
 2.5000000000000001,2.6000000000000001,2.7000000000000001,
 2.8000000000000001,2.9000000000000001,3.0000000000000001]
-- an error. To debug this try: debugmode(true);
```

Error in:

```
... =integrate(sqrt(diff(fx,t)^2+diff(fy,t)^2),t,0,t); s // elemen ...
^
```

```
>&kill(s); fx &= r*cos(s/r); fy &=r*sin(s/r); // definisi ulang persamaan parametrik terhadap s
>k &= trigsimp(sqrt(diff(fx,s,2)^2+diff(fy,s,2)^2)); $k // nilai kurvatur lingkaran dengan s
```

$$\frac{1}{r}$$

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

$$\begin{aligned}\kappa &= \frac{d\phi}{ds} = \frac{\frac{d\phi}{dt}}{\frac{ds}{dt}} \quad (\phi \text{ adalah sudut kemiringan garis singgung dan } s \text{ adalah panjang kurva}) \\ &= \frac{\frac{d\phi}{dt}}{\sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2}} = \frac{\frac{d\phi}{dt}}{\sqrt{x'(t)^2 + y'(t)^2}}.\end{aligned}$$

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 kurvurnya 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 p
```

$$k(x) = \frac{2a}{((2ax+b)^2 + 1)^{\frac{3}{2}}}$$

```
>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 p
```

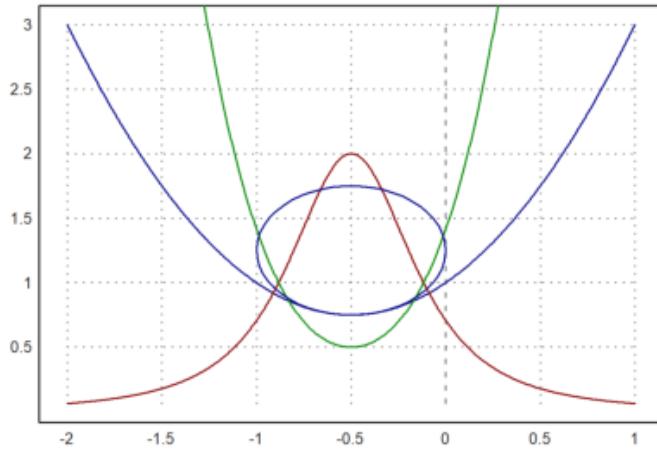
$$k(x) = \frac{2}{((2x+1)^2 + 1)^{\frac{3}{2}}}$$

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):
```



Untuk kurva yang dinyatakan dengan fungsi implisit

$$F(x, y) = 0$$

dengan turunan-turunan parsial

$$F_x = \frac{\partial F}{\partial x}, \quad F_y = \frac{\partial F}{\partial y}, \quad F_{xy} = \frac{\partial}{\partial y} \left(\frac{\partial F}{\partial x} \right), \quad F_{xx} = \frac{\partial}{\partial x} \left(\frac{\partial F}{\partial x} \right), \quad F_{yy} = \frac{\partial}{\partial y} \left(\frac{\partial F}{\partial y} \right),$$

berlaku

$$F_x dx + F_y dy = 0 \text{ atau } \frac{dy}{dx} = -\frac{F_x}{F_y},$$

sehingga kurvturnya adalah

$$\kappa = \frac{F_y^2 F_{xx} - 2F_x F_y F_{xy} + F_x^2 F_{yy}}{(F_x^2 + F_y^2)^{3/2}}.$$

(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)
```

$$2 a x + b$$

$$2 a$$

- 1

0

0

```
>function k(x) &= (Fy^2*Fx2-2*Fx*Fy*Fxy+Fx^2*Fyy)/(Fx^2+Fy^2)^(3/2); $'k(x)=k(x) // kurvat
```

$$k(x) = \frac{2a}{((2ax+b)^2 + 1)^{\frac{3}{2}}}$$

Hasilnya sama dengan sebelumnya yang menggunakan persamaan parabola biasa. **Latihan**

- Bukalah buku Kalkulus.
- Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi).
- Untuk setiap fungsi, tentukan anti turunannya (jika ada), hitunglah integral tentu dengan batas-batas yang menarik (Anda tentukan sendiri), seperti contoh-contoh tersebut.
- Lakukan hal yang sama untuk fungsi-fungsi yang tidak dapat diintegralkan (cari sedikitnya 3 fungsi).
- Gambar grafik fungsi dan daerah integrasinya pada sumbu koordinat yang sama.
- Gunakan integral tentu untuk mencari luas daerah yang dibatasi oleh dua kurva yang berpotongan di dua titik. (Cari dan gambar kedua kurva dan arsir (warnai) daerah yang dibatasi oleh keduanya.)
- Gunakan integral tentu untuk menghitung volume benda putar kurva $y = f(x)$ yang diputar mengelilingi sumbu x dari $x=a$ sampai $x=b$, yakni

$$V = \int_a^b \pi(f(x))^2 dx.$$

(Pilih fungsinya dan gambar kurva dan benda putar yang dihasilkan. Anda dapat mencari contoh-contoh bagaimana cara menggambar benda hasil perputaran suatu kurva.)

- Gunakan integral tentu untuk menghitung panjang kurva $y=f(x)$ dari $x=a$ sampai $x=b$ dengan menggunakan rumus:

$$S = \int_a^b \sqrt{1 + (f'(x))^2} dx.$$

(Pilih fungsi dan gambar kurvanya.)

- Apabila fungsi dinyatakan dalam koordinat kutub $x=f(r,t)$, $y=g(r,t)$, $r=h(t)$, $x=a$ bersesuaian dengan $t=t_0$ dan $x=b$ bersesuaian dengan $t=t_1$, maka rumus di atas akan menjadi:

$$S = \int_{t_0}^{t_1} \sqrt{x'(t)^2 + y'(t)^2} dt.$$

- Pilih beberapa kurva menarik (selain lingkaran dan parabola) dari buku kalkulus. Nyatakan setiap kurva tersebut dalam bentuk:

- a. koordinat Kartesius (persamaan $y=f(x)$)
- b. koordinat kutub ($r=r(\theta)$)
- c. persamaan parametrik $x=x(t)$, $y=y(t)$
- d. persamaan implisit $F(x,y)=0$

- Tentukan kurvatur masing-masing kurva dengan menggunakan keempat representasi tersebut (hasilnya harus sama).
- Gambarlah kurva asli, kurva kurvatur, kurva jari-jari lingkaran oskulasi, dan salah satu lingkaran oskulasinya.

Fungsi 1

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

$[0, 1.665833531718508 \times 10^{-4} r, 0.001330669204938795 r, 0.004479793338660443 r, 0.0105816576913495 r, 0.0205744613957969 r]$

```
>function intf(x) &= integrate(f(x),x); $intf(x)
```

$[0, 1.665833531718508 \times 10^{-4} r, 0.001330669204938795 r, 0.004479793338660443 r, 0.0105816576913495 r, 0.0205744613957969 r]$

```
>x=0:0.01:2.5-0.01; plot2d(x,f(x+0.01),>bar); plot2d("f(x)",0,2.5,>add):
```

Error : f(x) does not produce a real or column vector

Error generated by error() command

```
%ploteval:  
error(f$|" does not produce a real or column vector");  
adaptiveevalone:  
s=%ploteval(g$,t,args());  
Try "trace errors" to inspect local variables after errors.  
plot2d:  
dw/n,dw/n^2,dw/n,auto;args());
```

```
>t &= makelist(a,a,0,2.5-0.01,0.01);  
>fx &= makelist(f(t[i]+0.01),i,1,length(t));
```

Maxima output too long!

Error in:
fx &= makelist(f(t[i]+0.01),i,1,length(t)); ...
^

```
>>0.01*sum(f(x+0.01))
```

Syntax error in expression, or unfinished expression!
Error in:
>0.01*sum(f(x+0.01)) ...
^

>

Barisan dan Deret

(Catatan: bagian ini belum lengkap. Anda dapat membaca contoh-contoh penggunaan EMT dan Maxima untuk menghitung limit barisan, rumus jumlah parsial suatu deret, jumlah tak hingga suatu deret konvergen, dan sebagainya. Anda dapat mengeksplor contoh-contoh di EMT atau perbagai panduan penggunaan Maxima di software Maxima atau dari Internet.)

Barisan dapat didefinisikan dengan beberapa cara di dalam EMT, di antaranya:

- dengan cara yang sama seperti mendefinisikan vektor dengan elemen-elemen berurutan (menggunakan titik dua ":");
- menggunakan perintah "sequence" dan rumus barisan (suku ke -n);
- menggunakan perintah "iterate" atau "niterate";
- menggunakan fungsi Maxima "create_list" atau "makelist" untuk menghasilkan barisan simbolik;
- menggunakan fungsi biasa yang inputnya vektor atau barisan;
- menggunakan fungsi rekursif.

EMT menyediakan beberapa perintah (fungsi) terkait barisan, yakni:

- sum: menghitung jumlah semua elemen suatu barisan
- cumsum: jumlah kumulatif suatu barisan
- differences: selisih antar elemen-elemen berturutan

EMT juga dapat digunakan untuk menghitung jumlah deret berhingga maupun deret tak hingga, dengan menggunakan perintah (fungsi) "sum". Perhitungan dapat dilakukan secara numerik maupun simbolik dan eksak.

Berikut adalah beberapa contoh perhitungan barisan dan deret menggunakan EMT.

```
>1:10 // barisan sederhana
```

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

```
>1:2:30
```

```
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29]
```

Iterasi dan Barisan

EMT menyediakan fungsi iterate("g(x)", x0, n) untuk melakukan iterasi

$$x_{k+1} = g(x_k), \quad x_0 = x_0, \quad k = 1, 2, 3, \dots, n.$$

Berikut ini disajikan contoh-contoh penggunaan iterasi dan rekursi dengan EMT. Contoh pertama menunjukkan pertumbuhan dari nilai awal 1000 dengan laju pertambahan 5%, selama 10 periode.

```
>q=1.05; iterate("x*q",1000,n=10)'
```

```
1000
1050
1102.5
1157.63
1215.51
1276.28
```

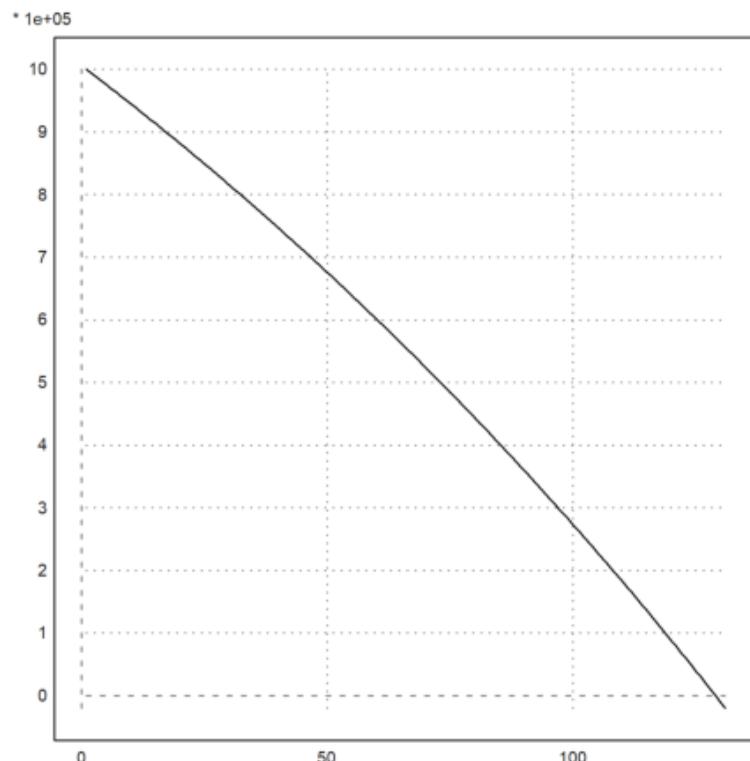
```

1340.1
1407.1
1477.46
1551.33
1628.89

```

Contoh berikutnya memperlihatkan bahaya menabung di bank pada masa sekarang! Dengan bunga tabungan sebesar 6% per tahun atau 0.5% per bulan dipotong pajak 20%, dan biaya administrasi 10000 per bulan, tabungan sebesar 1 juta tanpa diambil selama sekitar 10 tahunan akan habis diambil oleh bank!

```
>r=0.005; plot2d(iterate("(1+0.8*r)*x-10000",1000000,n=130)):
```



Silakan Anda coba-coba, dengan tabungan minimal berapa agar tidak akan habis diambil oleh bank dengan ketentuan bunga dan biaya administrasi seperti di atas.

Berikut adalah perhitungan minimal tabungan agar aman di bank dengan bunga sebesar r dan biaya administrasi a , pajak bunga 20%.

```
>$solve(0.8*r*A-a,A), $% with [r=0.005, a=10]
```

$$[A = 2500.0]$$

images/Shalih Abdillah_22305144009_ApKom5-230-lar

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, $x[1], \dots, x[n-1]$ yang diketahui.

Berikut adalah contoh barisan Fibonacci.

$$x_n = x_{n-1} + x_{n-2}, \quad x_1 = 1, \quad x_2 = 1$$

```
>sequence("x[n-1]+x[n-2]",[1,1],15)
```

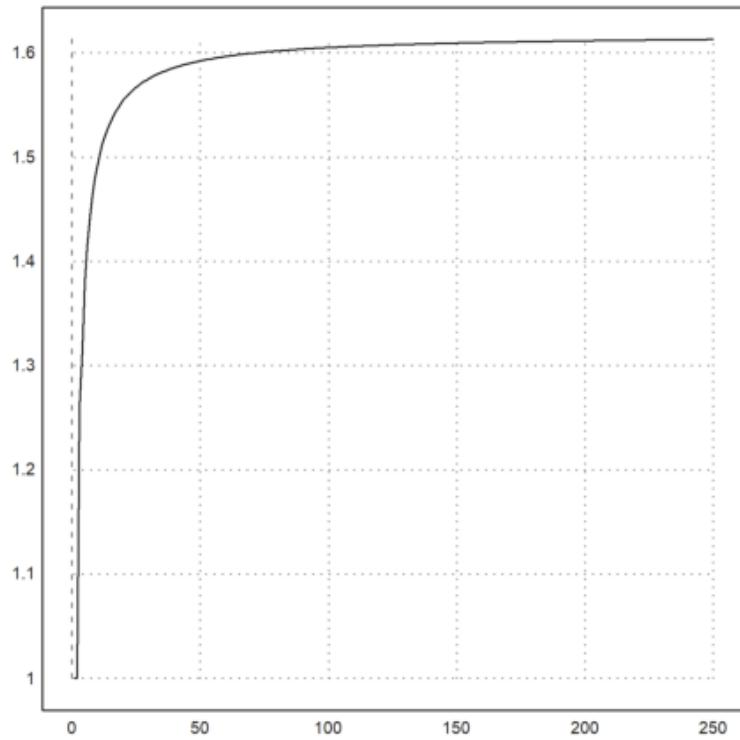
```
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610]
```

Barisan Fibonacci memiliki banyak sifat menarik, salah satunya adalah akar pangkat ke-n suku ke-n akan konvergen ke pecahan emas:

```
>$(1+sqrt(5))/2=float((1+sqrt(5))/2)
```

$$\frac{\sqrt{5} + 1}{2} = 1.618033988749895$$

```
>plot2d(sequence("x[n-1]+x[n-2]",[1,1],250)^(1/(1:250))):
```



Barisan yang sama juga dapat dihasilkan dengan menggunakan loop.

```
>x=ones(500); for k=3 to 500; x[k]=x[k-1]+x[k-2]; end;
```

Rekursi dapat dilakukan dengan menggunakan rumus yang tergantung pada semua elemen sebelumnya. Pada contoh berikut, elemen ke-n merupakan jumlah (n-1) elemen sebelumnya, dimulai dengan 1 (elemen ke-1). Jelas, nilai elemen ke-n adalah $2^{(n-2)}$, untuk n=2, 4, 5,

```
>sequence("sum(x)",1,10)
```

```
[1, 1, 2, 4, 8, 16, 32, 64, 128, 256]
```

Selain menggunakan ekspresi dalam x dan n, kita juga dapat menggunakan fungsi. Pada contoh berikut, digunakan iterasi

$$x_n = A \cdot x_{n-1},$$

dengan A suatu matriks 2x2, dan setiap x[n] merupakan matriks/vektor 2x1.

```
>A=[1,1;1,2]; function suku(x,n) := A.x[,n-1]
>sequence("suku",[1;1],6)
```

Real 2 x 6 matrix

1	2	5	13	...
1	3	8	21	...

Hasil yang sama juga dapat diperoleh dengan menggunakan fungsi perpangkatan matriks "matrixpower()". Cara ini lebih cepat, karena hanya menggunakan perkalian matriks sebanyak $\log_2(n)$.

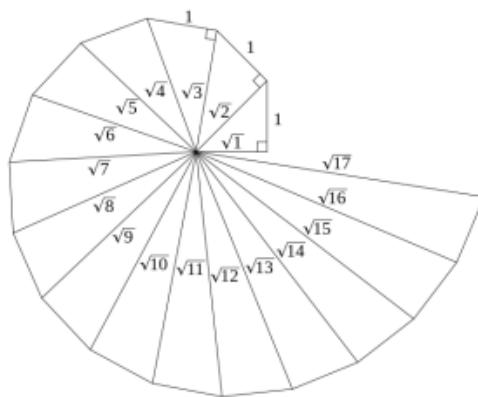
$$x_n = A \cdot x_{n-1} = A^2 \cdot x_{n-2} = A^3 \cdot x_{n-3} = \dots = A^{n-1} \cdot x_1.$$

```
>sequence("matrixpower(A, n) . [1;1]", 1, 6)
```

Real 2 x 6 matrix

1	5	13	34	...
1	8	21	55	...

Spiral Theodorus



Spiral Theodorus (spiral segitiga siku-siku) dapat digambar secara rekursif. Rumus rekursifnya adalah:

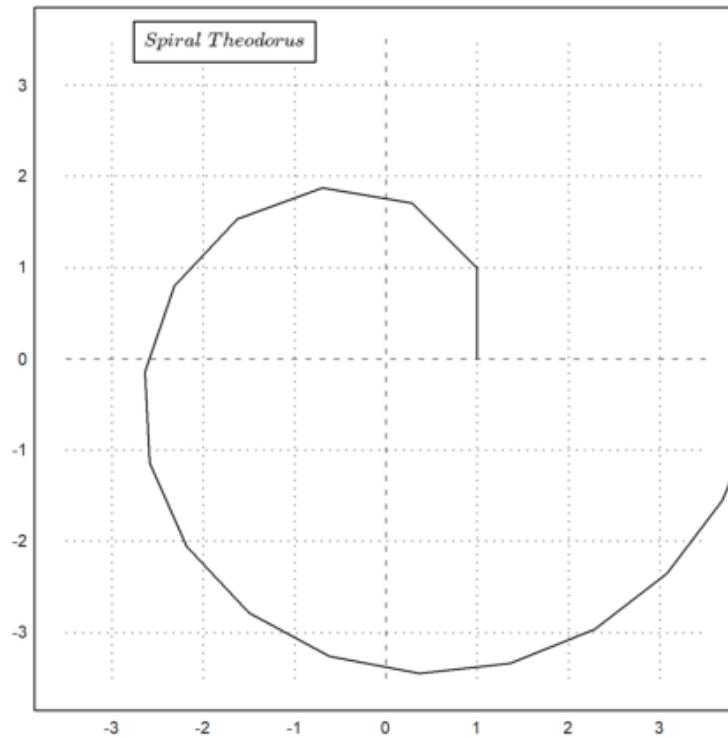
$$x_n = \left(1 + \frac{i}{\sqrt{n-1}}\right) x_{n-1}, \quad x_1 = 1,$$

yang menghasilkan barisan bilangan kompleks.

```
>function g(n) := 1+I/sqrt(n)
```

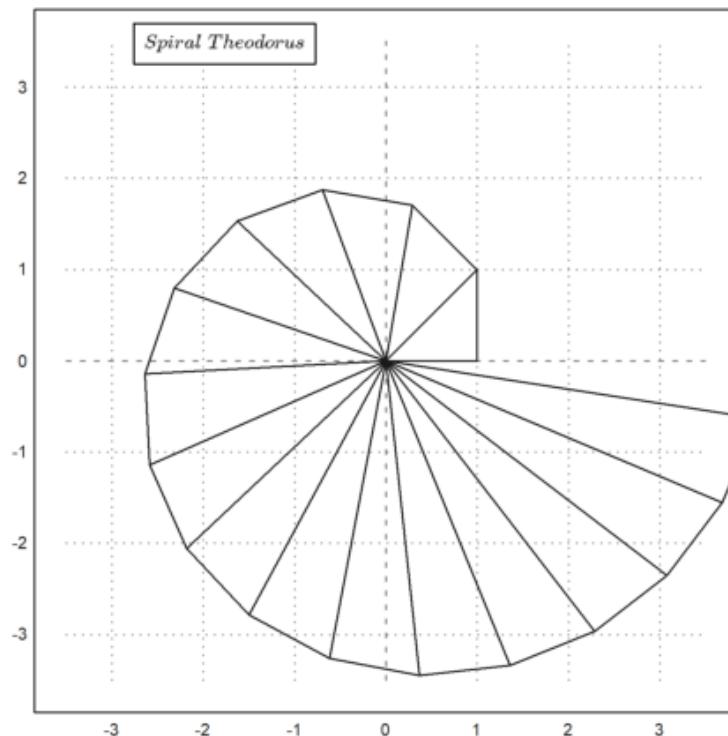
Rekursinya dapat dijalankan sebanyak 17 untuk menghasilkan barisan 17 bilangan kompleks, kemudian di-gambar bilangan-bilangan kompleksnya.

```
>x=sequence("g(n-1)*x[n-1]",1,17); plot2d(x,r=3.5); textbox(latex("Spiral\ Theodorus"),0.4
```



Selanjutnya dihubungkan titik 0 dengan titik-titik kompleks tersebut menggunakan loop.

```
>for i=1:cols(x); plot2d([0,x[i]],>add); end;
```



```
>
```

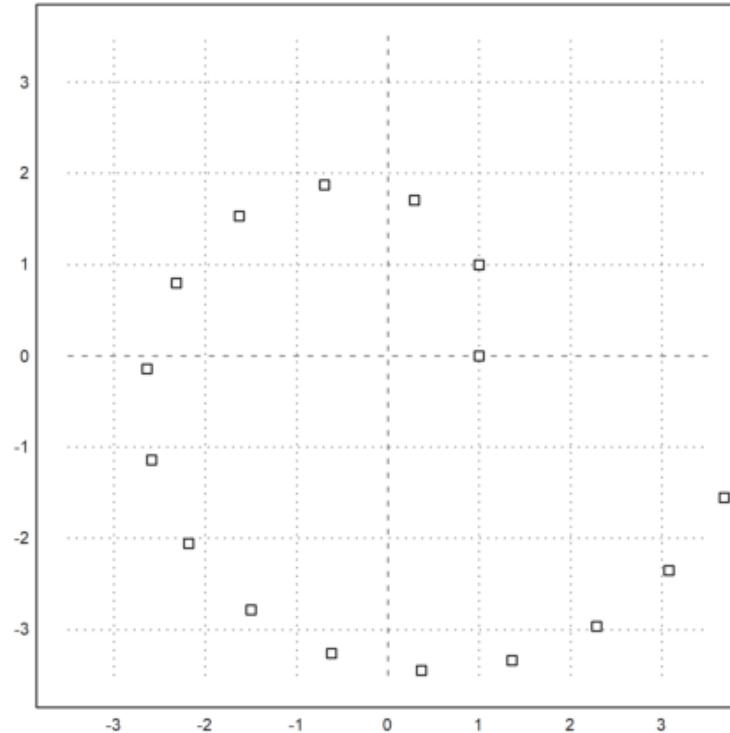
Spiral tersebut juga dapat didefinisikan menggunakan fungsi rekursif, yang tidak memerlukan indeks dan bilangan kompleks. Dalam hal ini digunakan vektor kolom pada bidang.

```
>function gstep (v) ...
```

```
w=[-v[2];v[1]];
return v+w/norm(w);
endfunction
```

Jika dilakukan iterasi 16 kali dimulai dari [1;0] akan didapatkan matriks yang memuat vektor-vektor dari setiap iterasi.

```
>x=iterate("gstep",[1;0],16); plot2d(x[1],x[2],r=3.5,>points):
```



Kekonvergenan

Terkadang kita ingin melakukan iterasi sampai konvergen. Apabila iterasinya tidak konvergen setelah ditunggu lama, Anda dapat menghentikannya dengan menekan tombol [ESC].

```
>iterate("cos(x)",1) // iterasi x(n+1)=cos(x(n)), dengan x(0)=1.
```

0.739085133216

Iterasi tersebut konvergen ke penyelesaian persamaan

$$x = \cos(x).$$

Iterasi ini juga dapat dilakukan pada interval, hasilnya adalah barisan interval yang memuat akar tersebut.

```
>hasil := iterate("cos(x)", ~1, 2~) //iterasi x(n+1)=cos(x(n)), dengan interval awal (1, 2)
```

$\sim 0.739085133211, 0.7390851332133\sim$

Jika interval hasil tersebut sedikit diperlebar, akan terlihat bahwa interval tersebut memuat akar persamaan $x=\cos(x)$.

```
>h=expand(hasil,100), cos(h) << h
```

$\sim 0.73908513309, 0.73908513333\sim$

1

Iterasi juga dapat digunakan pada fungsi yang didefinisikan.

```
>function f(x) := (x+2/x)/2
```

Iterasi $x(n+1)=f(x(n))$ akan konvergen ke akar kuadrat 2.

```
>iterate("f", 2), sqrt(2)
```

1.41421356237

1.41421356237

Jika pada perintah iterate diberikan tambahan parameter n, maka hasil iterasinya akan ditampilkan mulai dari iterasi pertama sampai ke-n.

```
>iterate("f", 2, 5)
```

[2, 1.5, 1.41667, 1.41422, 1.41421, 1.41421]

Untuk iterasi ini tidak dapat dilakukan terhadap interval.

```
>niterate("f", ~1, 2~, 5)
```

[~1, 2~, ~1, 2~, ~1, 2~, ~1, 2~, ~1, 2~, ~1, 2~]

Perhatikan, hasil iterasinya sama dengan interval awal. Alasannya adalah perhitungan dengan interval bersifat terlalu longgar. Untuk meningkatkan 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~)
```

$\sim 1.41421356236, 1.41421356239\sim$

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~	...
~4.999999999999911, 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 dipercepat dengan percepatan Steffenson:

```
>steffenson("sqrt(x)",2,10)
```

```
[1.04888, 1.00028, 1, 1]
```

Iterasi menggunakan Loop yang ditulis Langsung

Berikut adalah beberapa contoh penggunaan loop untuk melakukan iterasi yang ditulis langsung pada baris perintah.

```
>x=2; repeat x=(x+2/x)/2; until x^2~=2; end; x,
```

```
1.41421356237
```

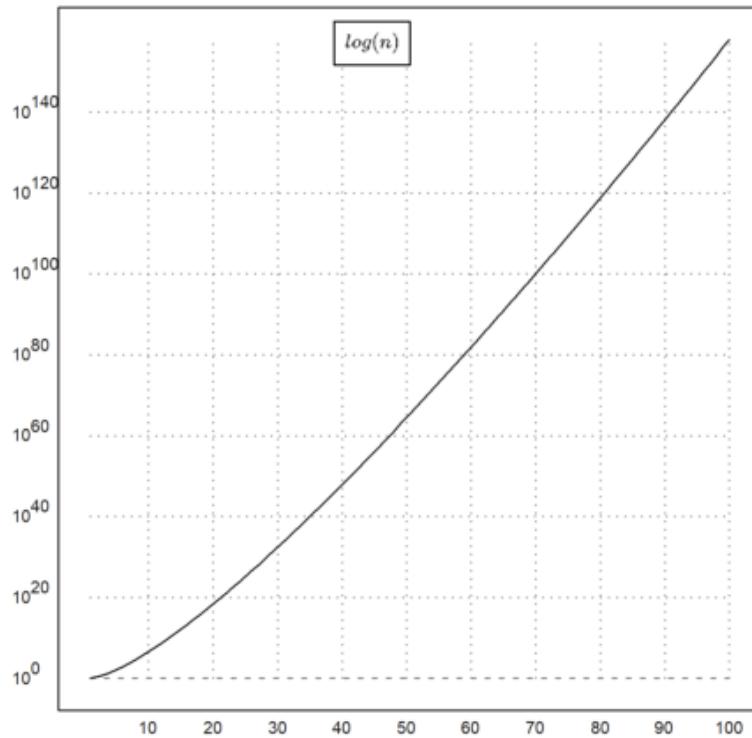
Penggabungan matriks menggunakan tanda "|" dapat digunakan untuk menyimpan semua hasil iterasi.

```
>v=[1]; for i=2 to 8; v=v|(v[i-1]*i); end; v,
```

```
[1, 2, 6, 24, 120, 720, 5040, 40320]
```

hasil iterasi juga dapat disimpan pada vektor yang sudah ada.

```
>v=ones(1,100); for i=2 to cols(v); v[i]=v[i-1]*i; end; ...  
>plot2d(v,logplot=1); textbox(latex(&log(n)),x=0.5):
```



```
>A =[0.5,0.2;0.7,0.1]; b=[2;2]; ...
>x=[1;1]; repeat xnew=A.x-b; until all(xnew~=x); x=xnew; end; ...
>x,
```

```
-7.09677
-7.74194
```

Iterasi di dalam Fungsi

Fungsi atau program juga dapat menggunakan iterasi dan dapat digunakan untuk melakukan iterasi. Berikut adalah beberapa contoh iterasi di dalam fungsi.

Contoh berikut adalah suatu fungsi untuk menghitung berapa lama suatu iterasi konvergen. Nilai fungsi tersebut adalah hasil akhir iterasi dan banyak iterasi sampai konvergen.

```
>function map hiter(f$,x0) ...
```

```
x=x0;
maxiter=0;
repeat
  xnew=f$(x);
  maxiter=maxiter+1;
  until xnew~=x;
  x=xnew;
end;
return maxiter;
endfunction
```

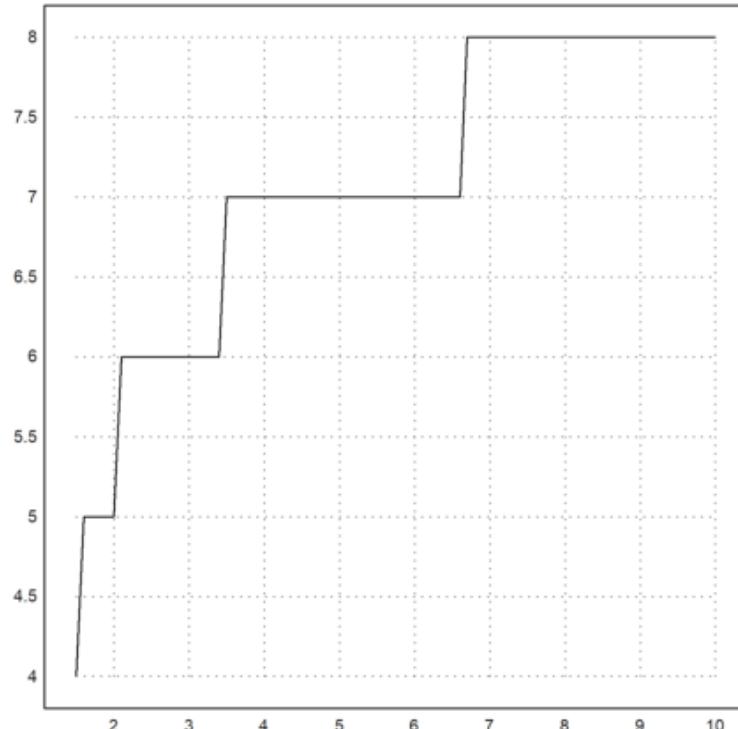
Misalnya, berikut adalah iterasi untuk mendapatkan hampiran akar kuadrat 2, cukup cepat, konvergen pada iterasi ke-5, jika dimulai dari hampiran awal 2.

```
>hiter("(x+2/x)/2", 2)
```

5

Karena fungsinya didefinisikan menggunakan "map". maka nilai awalnya dapat berupa vektor.

```
>x=1.5:0.1:10; hasil=hiter("(x+2/x)/2", x); ...
> plot2d(x,hasil):
```



Dari gambar di atas terlihat bahwa kekonvergenan iterasinya semakin lambat, untuk nilai awal semakin besar, namun penambahannya tidak kontinu. Kita dapat menemukan kapan maksimum iterasinya bertambah.

```
>hasil[1:10]
```

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

```
>x[nonzeros(differences(hasil))]
```

```
[1.5, 2, 3.4, 6.6]
```

maksimum iterasi sampai konvergen meningkat pada saat nilai awalnya 1.5, 2, 3.4, dan 6.6.
Contoh berikutnya adalah metode Newton pada polinomial kompleks berderajat 3.

```
>p &= x^3-1; newton &= x-p/diff(p,x); $newton
```

Maxima said:

```
diff: second argument must be a variable; found errexpl  
-- an error. To debug this try: debugmode(true);
```

Error in:

```
p &= x^3-1; newton &= x-p/diff(p,x); $newton ...  
^
```

Selanjutnya didefinisikan fungsi untuk melakukan iterasi (aslinya 10 kali).

```
>function iterasi(f$,x,n=10) ...
```

```
loop 1 to n; x=f$(x); end;  
return x;  
endfunction
```

Kita mulai dengan menentukan titik-titik grid pada bidang kompleksnya.

```
>r=1.5; x=linspace(-r,r,501); Z=x+I*x'; W=iterasi(newton,Z);
```

Function newton needs at least 3 arguments!

```
Use: newton (f$: call, df$: call, x: scalar complex {, y: number, eps: none})
```

Error in:

```
... x=linspace(-r,r,501); Z=x+I*x'; W=iterasi(newton,Z); ...  
^
```

Berikut adalah akar-akar polinomial di atas.

```
>z=&solve(p)()
```

Maxima said:

```
solve: more equations than unknowns.
```

Unknowns given :

```
[r]
```

Equations given:

```
errexpl
```

```
-- an error. To debug this try: debugmode(true);
```

Error in:

```
z=&solve(p)() ...  
^
```

Untuk menggambar hasil iterasinya, dihitung jarak dari hasil iterasi ke-10 ke masing-masing akar, kemudian digunakan untuk menghitung warna yang akan digambar, yang menunjukkan limit untuk masing-masing nilai awal.

Fungsi plotrgb() menggunakan jendela gambar terkini untuk menggambar warna RGB sebagai matriks.

```
>C=rgb(max(abs(W-z[1]),1),max(abs(W-z[2]),1),max(abs(W-z[3]),1)); ...  
> plot2d(None,-r,r,-r,r); plotrgb(C);
```

```
Variable W not found!
Error in:
C=rgb(max(abs(W-z[1])),1),max(abs(W-z[2]),1),max(abs(W-z[3]),1) ...  
^
```

Iterasi Simbolik

Seperti sudah dibahas sebelumnya, untuk menghasilkan barisan ekspresi simbolik dengan Maxima dapat digunakan fungsi makelist().

```
>&powerdisp:true // untuk menampilkan deret pangkat mulai dari suku berpangkat terkecil
```

true

```
>deret &= makelist(taylor(exp(x),x,0,k),k,1,3); $deret // barisan deret Taylor untuk e^x
```

```
Maxima said:
taylor: 0.1539740213994798*r cannot be a variable.
-- an error. To debug this try: debugmode(true);
```

```
Error in:
deret &= makelist(taylor(exp(x),x,0,k),k,1,3); $deret // baris ...
^
```

Untuk mengubah barisan deret tersebut menjadi vektor string di EMT digunakan fungsi mxm2str(). Selanjutnya, vektor string/ekspresi hasilnya dapat digambar seperti menggambar vektor ekspresi 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 ters
```

```
Maxima said:
length: argument cannot be a symbol; found deret
-- an error. To debug this try: debugmode(true);

mxmeval:
    return evaluate(mxm(s));
Try "trace errors" to inspect local variables after errors.
mxm2str:
    n=mxmeval("length(VVV)");
```

Selain cara di atas dapat juga dengan cara menggunakan indeks pada vektor/list yang dihasilkan.

```
>$deret[3]
```

*deret*₃

```
>plot2d(["exp(x)",&deret[1],&deret[2],&deret[3]],0,3,color=1:4):
```

```

deret is not a variable!
Error in expression: deret[1]
%ploteval:
y0=f$(x[1],args());
Try "trace errors" to inspect local variables after errors.
plot2d:
u=u_(%ploteval(xx[#],t,args()));

```

```
> $sum(sin(k*x)/k,k,1,5)
```

$$\frac{1333077512 r}{5} \left[\sin\left(0.\underline{1165626293080298} r\right) + 10 \frac{\sin\left(0.29810032390366667 r\right)}{2} + \frac{14883 \sin\left(0.4785000072032980105780 r\right)}{3} + \frac{788352 \sin\left(0.66663000072032980105780 r\right)}{3} + \frac{321194 \sin\left(0.85472000072032980105780 r\right)}{4} + \dots \right]$$

Berikut adalah cara menggambar kurva

$$y = \sin(x) + \frac{\sin 3x}{3} + \frac{\sin 5x}{5} + \dots$$

```
> plot2d(&sum(sin((2*k+1)*x)/(2*k+1),k,0,20),0,2pi):
```

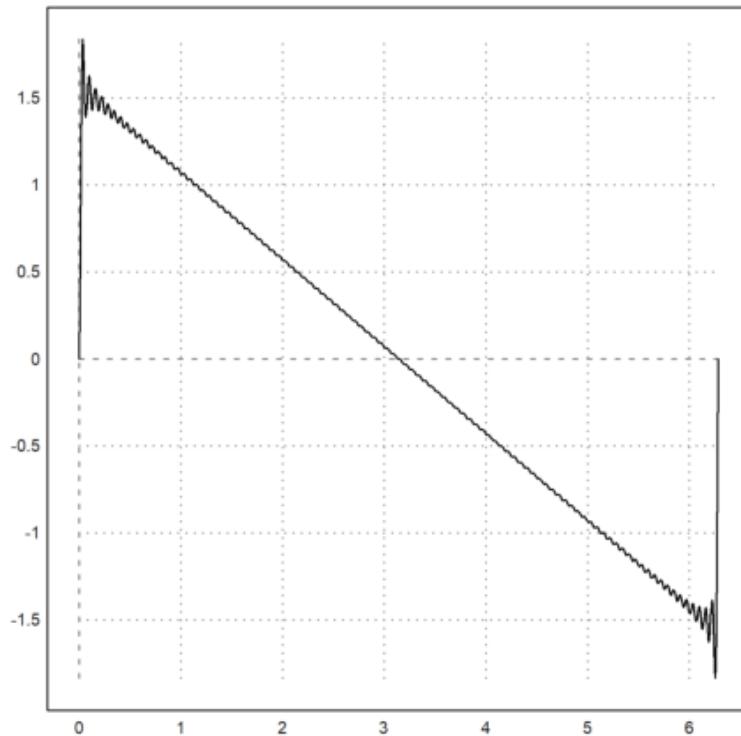
Maxima output too long!

```
Error in:
plot2d(&sum(sin((2*k+1)*x)/(2*k+1),k,0,20),0,2pi): ...
```

Hal serupa juga dapat dilakukan dengan menggunakan matriks, misalkan kita akan menggambar kurva

$$y = \sum_{k=1}^{100} \frac{\sin(kx)}{k}, \quad 0 \leq x \leq 2\pi.$$

```
>x=linspace(0,2pi,1000); k=1:100; y=sum(sin(k*x')/k)'; plot2d(x,y):
```



Tabel Fungsi

Terdapat cara menarik untuk menghasilkan barisan dengan ekspresi Maxima. Perintah mxmtable() berguna untuk menampilkan dan menggambar barisan dan menghasilkan barisan sebagai vektor kolom. Sebagai contoh berikut adalah barisan turunan ke-n x^n di $x=1$.

```
>mxmtable("diffat(x^x,x=1,n)","n",1,8,frac=1);
```

Maxima said:

```
diff: second argument must be a variable; found errexp1
#0: diffat(expr=[0,1.66665833335744e-7*r,1.33330666692022e-6*r,4.499797504338432e-6*r,1.
-- an error. To debug this try: debugmode(true);
```

%mxmtable:

```
    return mxm("@expr,@var=@value")();
```

Try "trace errors" to inspect local variables after errors.

mxmtable:

```
    y[#,1]=%mxmtable(expr,var,x[#]);
```

```
>$' sum(k, k, 1, n) = factor(ev(sum(k, k, 1, n),simpsum=true)) // simpsum:menghitung deret
```

$$\sum_{k=1}^n k = \frac{n(n+1)}{2}$$

```
>$' 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 e
```

$$\sum_{x=1}^{\infty} \frac{1}{x^2} = \frac{\pi^2}{6}$$

```
> $' 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))
```

$$\sum_{k=1}^{\infty} \frac{(-1)^k}{-1 + 2k} = \sum_{k=1}^{\infty} \frac{(-1)^k}{-1 + 2k}$$

```
> $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 ...
```

Answering "Is $-94914474571+15819\sqrt{r}$ positive, negative or zero?" with "positive"
Maxima said:
sum: sum is divergent.
-- an error. To debug this try: debugmode(true);

Error in:
... k, 0, inf)=ev(sum(a*x^k, k, 0, inf), simpsum=true), &forget(abs ...
^

Deret geometri tak hingga, dengan asumsi rasional antara -1 dan 1.

```
> $' sum(x^k/k!, k, 0, inf)=ev(sum(x^k/k!, k, 0, inf), simpsum=true)
```

$$\left[0, \sum_{k=0}^{\infty} \frac{(1.66665833335744 \times 10^{-7})^k r^k}{k!}, \sum_{k=0}^{\infty} \frac{(1.33330666692022 \times 10^{-6})^k r^k}{k!}, \sum_{k=0}^{\infty} \frac{(4.499797504338432 \times 10^{-6})^k r^k}{k!}, \sum_{k=0}^{\infty} \frac{(1.00000000000000 \times 10^{-6})^k r^k}{k!} \right]$$

```
>$limit(sum(x^k/k!, k, 0, n), n, inf)
```

$$\left[0, \lim_{n \rightarrow \infty} \sum_{k=0}^n \frac{(1.66665833335744 \times 10^{-7})^k r^k}{k!}, \lim_{n \rightarrow \infty} \sum_{k=0}^n \frac{(1.33330666692022 \times 10^{-6})^k r^k}{k!}, \lim_{n \rightarrow \infty} \sum_{k=0}^n \frac{(4.499797504338432 \times 10^{-1})^k r^k}{k!} \right]$$

```
>function d(n) &= sum(1/(k^2-k), k, 2, n); $'d(n)=d(n)
```

$$d(n) = \sum_{k=2}^n \frac{1}{-k + k^2}$$

```
>$d(10)=ev(d(10), simpsum=true)
```

$$\sum_{k=2}^{10} \frac{1}{-k + k^2} = \frac{9}{10}$$

```
>$d(100)=ev(d(100), simpsum=true)
```

$$\sum_{k=2}^{100} \frac{1}{-k + k^2} = \frac{99}{100}$$

Deret Taylor

Deret Taylor suatu fungsi f yang diferensiabel sampai tak hingga di sekitar $x=a$ adalah:

$$f(x) = \sum_{k=0}^{\infty} \frac{(x-a)^k f^{(k)}(a)}{k!}.$$

```
>$'e^x=taylor(exp(x),x,0,10) // deret Taylor e^x di sekitar x=0, sampai suku ke-11
```

Maxima said:

```
taylor: 2.858879991940135*r cannot be a variable.  
-- an error. To debug this try: debugmode(true);
```

Error in:

```
$'e^x=taylor(exp(x),x,0,10) // deret Taylor e^x di sekitar x= ...  
^
```

```
>$'log(x)=taylor(log(x),x,1,10)// deret log(x) di sekitar x=1
```

Maxima said:

```
log: encountered log(0).  
-- an error. To debug this try: debugmode(true);
```

Error in:

```
$'log(x)=taylor(log(x),x,1,10)// deret log(x) di sekitar x=1 ...  
^
```

BAB 6

KB PEKAN 8: MENGGUNAKAN EMT UNTUK GEOMETRI

article
eumat

Visualisasi dan Perhitungan Geometri dengan EMT

Nama : Shalih Abdillah
NIM : 22305144009
Kelas : Matematika B 2022

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

```
>load geometry
```

Numerical and symbolic geometry.

Fungsi-fungsi Geometri

Fungsi-fungsi untuk Menggambar Objek Geometri:

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

```
setPlotRange(r): pusat bidang koordinat (0,0) dan batas-batas  
sumbu-x dan y adalah -r sd r
```

```

plotPoint (P, "P"): menggambar titik P dan diberi label "P"
plotSegment (A,B, "AB", d): menggambar ruas garis AB, diberi label
"AB" sejauh d

```

```

plotLine (g, "g", d): menggambar garis g diberi label "g" sejauh d
plotCircle (c,"c",v,d): Menggambar lingkaran c dan diberi label "c"
plotLabel (label, P, V, d): menuliskan label pada posisi P

```

Fungsi-fungsi Geometri Analitik (numerik maupun simbolik):

```

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

```

$ax+by=c$.

```

lineWithDirection(A,v): garis melalui A searah vektor v
getLineDirection(g): vektor arah (gradien) garis g
getNormal(g): vektor normal (tegak lurus) garis g
getPointOnLine(g): titik pada garis g
perpendicular(A, g): garis melalui A tegak lurus garis g
parallel (A, g): garis melalui A sejajar garis g
lineIntersection(g, h): titik potong garis g dan h
projectToLine(A, g): proyeksi titik A pada garis g
distance(A, B): jarak titik A dan B
distanceSquared(A, B): kuadrat jarak A dan B
quadrance(A, B): kuadrat jarak A dan B
areaTriangle(A, B, C): luas segitiga ABC
computeAngle(A, B, C): besar sudut <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 lingkaran c
circleCircleIntersections (c1, c2): titik potong lingkaran c1 dan

```

c2

```
planeThrough(A, B, C): bidang melalui titik A, B, C
```

Fungsi-fungsi Khusus Untuk Geometri Simbolik:

```

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

```

```
sisi positif (kanan/atas) garis  
quad(A,B) : kuadrat jarak AB  
spread(a,b,c) : Spread segitiga dengan panjang sisi-sisi a,b,c, yakni
```

sin(alpha)^2 dengan

```
alpha sudut yang menghadap sisi a.  
crosslaw(a,b,c,sa) : persamaan 3 quads dan 1 spread pada segitiga  
dengan panjang sisi a, b, c.
```

```
triplespread(sa,sb,sc) : persamaan 3 spread sa,sb,sc yang memebntuk  
suatu segitiga
```

```
doublespread(sa) : Spread sudut rangkap Spread 2*phi, dengan  
sa=sin(phi)^2 spread a.
```

Contoh 1: Luas, Lingkaran Luar, Lingkaran Dalam Segitiga

Untuk menggambar objek-objek geometri, langkah pertama adalah menentukan rentang sumbu-sumbu koordinat. Semua objek geometri akan digambar pada satu bidang koordinat, sampai didefinisikan bidang koordinat yang baru.

```
>setPlotRange(-0.5,2.5,-0.5,2.5); // mendefinisikan bidang koordinat baru
```

Sekarang atur tiga poin dan plot.

```
>A=[1,0]; plotPoint(A,"A"); // definisi dan gambar tiga titik  
>B=[0,1]; plotPoint(B,"B");  
>C=[2,2]; plotPoint(C,"C");
```

Lalu tiga segmen.

```
>plotSegment(A,B,"c"); // c=AB  
>plotSegment(B,C,"a"); // a=BC  
>plotSegment(A,C,"b"); // b=AC
```

Fungsi geometri meliputi fungsi untuk membuat garis dan lingkaran. Format untuk garis adalah [a, b, c], yang merepresentasikan garis dengan persamaan $ax + by = c$.

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

$[-1, 2, 2]$

Hitung garis tegak lurus melalui A pada BC.

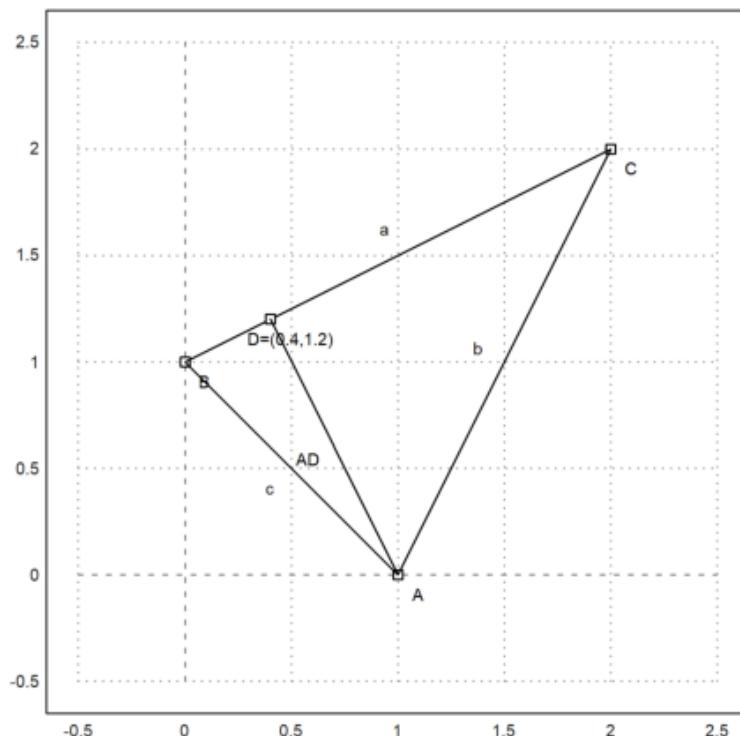
```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan perpotongannya dengan BC.

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

Plot itu.

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



Hitung luas ABC:

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

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

1.5

Compare with determinant formula.

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

1.5

Cara lain menghitung luas segitiga ABC:

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

1.5

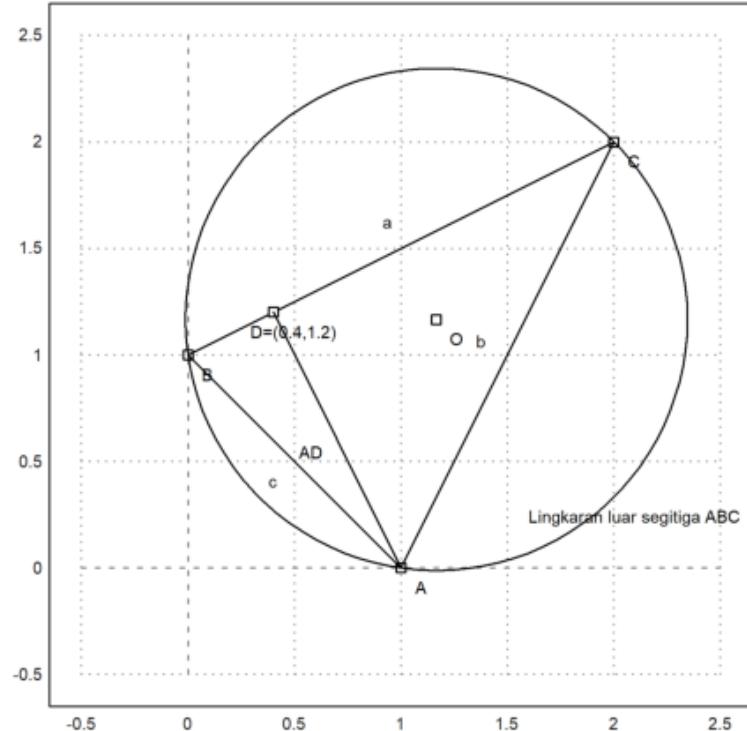
Sudut di C

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

$36^\circ 52' 11.63''$

Sekarang lingkaran sirkit segitiga.

```
>c=circleThrough(A,B,C); // lingkaran luar segitiga ABC  
>R=getCircleRadius(c); // jari2 lingkaran luar  
>O=getCircleCenter(c); // titik pusat lingkaran c  
>plotPoint(O,"O"); // gambar titik "O"  
>plotCircle(c,"Lingkaran luar segitiga ABC");
```



Tampilkan koordinat titik pusat dan jari-jari lingkaran luar.

```
>O, R
```

```
[1.16667, 1.16667]  
1.17851130198
```

Sekarang akan digambar lingkaran dalam segitiga ABC. Titik pusat lingkaran dalam adalah titik potong garis-garis bagi sudut.

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

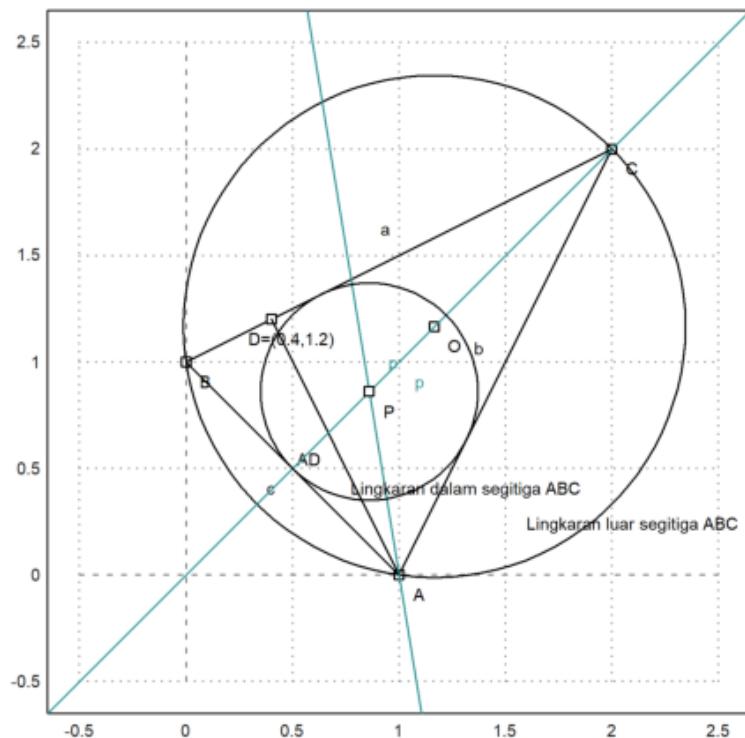
[0.86038, 0.86038]

Tambahkan semua ke plot.

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

0.509653732104

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



Latihan

1. Tentukan ketiga titik singgung lingkaran dalam dengan sisi-sisi segitiga ABC.
2. Gambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut. Merupakan segitiga apakah itu?
3. Hitung luas segitiga tersebut.
4. Tunjukkan bahwa garis bagi sudut yang ke tiga juga melalui titik pusat lingkaran dalam.
5. Gambar jari-jari lingkaran dalam.

6. Hitung luas lingkaran luar dan luas lingkaran dalam segitiga ABC. Adakah hubungan antara luas kedua lingkaran tersebut dengan luas segitiga ABC?

Jawab :

1.

```
>s=lineThrough(B,C)
```

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

```
>m=circleWithCenter(P,r) ...
>S=lineCircleIntersections(s,m) ...
```

Titik singgung garis AC dengan lingkaran dalam.

```
>p=lineThrough(A,C) ...
>Q=lineCircleIntersections(p,m) ...
>L=lineCircleIntersections(q,m)
```

Commands must be separated by semicolon or comma!

Found: S=lineCircleIntersections(s,m) p=lineThrough(A,C) Q=lineCircleIntersections(p,m)
You can disable this in the Options menu.

Error in:

```
m=circleWithCenter(P,r) S=lineCircleIntersections(s,m) p=lineT ...
^
```

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.

```
>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 ...
>k=angleBisector(A,B,C) ...
>color(2); plotLine(k);
```

4. Gambar jari-jari lingkaran dalam.

```
>plotSegment(P,L,"r");
```

Variable or function S not found.

Error in:

```
plotSegment(S,Q,"a"); plotSegment(S,L,"b"); plotSegment(L,Q,"c") ...
^
```

Contoh 2: Geometri Smbolik

Kita dapat menghitung geometri tepat dan simbolis menggunakan Maxima.

Geometri file.e menyediakan fungsi yang sama (dan lebih banyak lagi) di Maxima. Namun, sekarang kita dapat menggunakan perhitungan simbolik.

```
>A &= [1,0]; B &= [0,1]; C &= [2,2]; // menentukan tiga titik A, B, C
```

Fungsi garis dan lingkaran bekerja seperti fungsi Euler, tetapi menyediakan penghitungan simbolik.

```
>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
```

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

 images/Shalih Abdillah_22305144009_ApKom6-006-large

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

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

 images/Shalih Abdillah_22305144009_ApKom6-008-large

```
>$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
```

$$\begin{bmatrix} 2 & 6 \\ - & - \\ 5 & 5 \end{bmatrix}$$

```
>$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
```

$$1.178511301977579$$

images/Shalih Abdillah_22305144009_ApKom6-014-lar

```
>$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
```

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

```
>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.

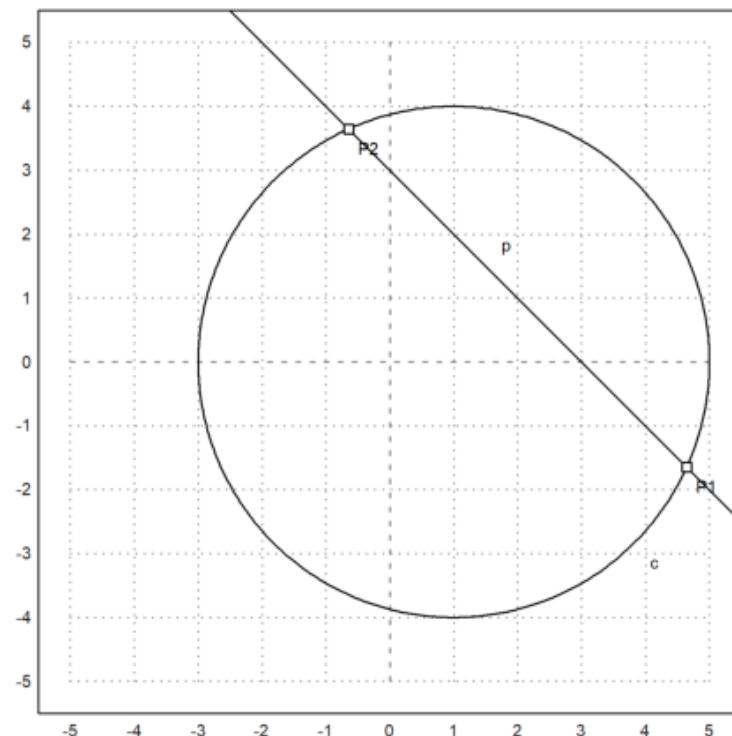
```
>A &:= [1,0]; c=circleWithCenter(A,4);  
>B &:= [1,2]; C &:= [2,1]; l=lineThrough(B,C);  
>setPlotRange(5); plotCircle(c); plotLine(l);
```

Perpotongan garis dengan lingkaran mengembalikan dua titik dan jumlah titik perpotongan.

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

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

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



Hal yang sama di Maxima.

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

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

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

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

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

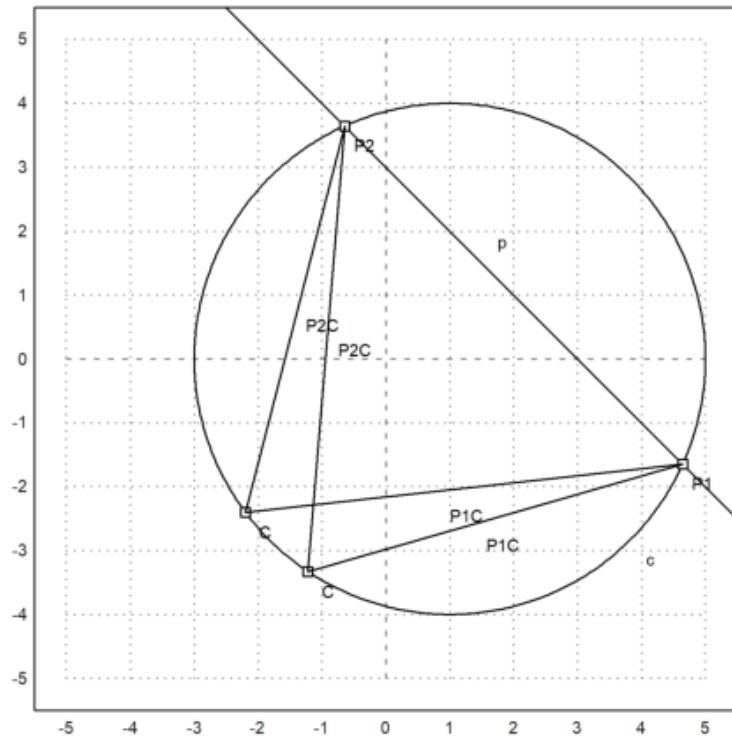
```
>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);  
>deprint(computeAngle(P1,C,P2))
```

```
69°17'42.68''
```

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

```
69°17'42.68''
```

```
>insimg;
```

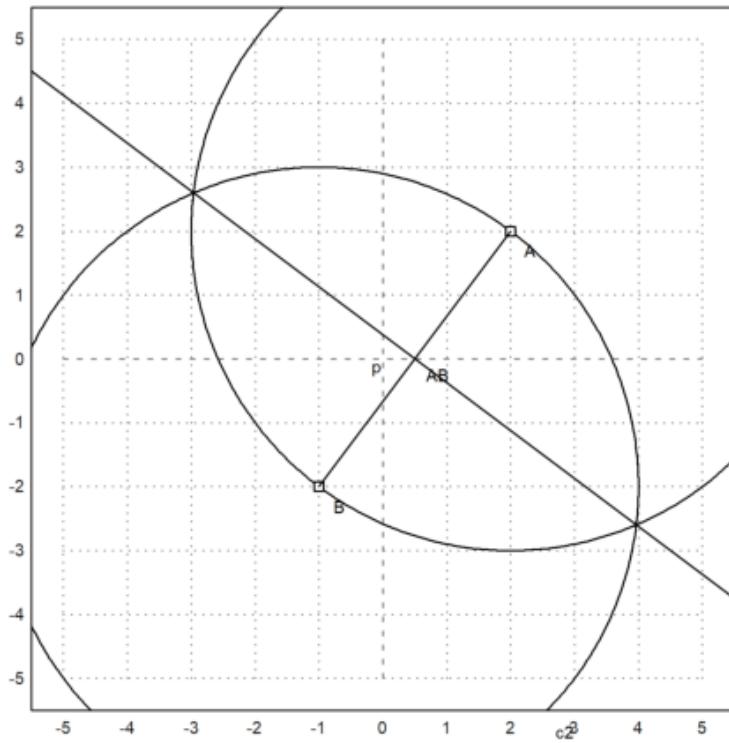


Garis Sumbu

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

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

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



Selanjutnya, kita melakukan hal yang sama di Maxima dengan koordinat umum.

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

Persamaan untuk persimpangan cukup terlibat. Tapi kita bisa menyederhanakan, jika kita menyelesaikan y.

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

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

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

```
>$solve(getLineEquation(middlePerpendicular(A,B),x,y),y)
```

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

```
>h &= getLineEquation(lineThrough(A,B),x,y);
>$solve(h,y)
```

$$\left[y = \frac{(b_2 - a_2)x - a_1b_2 + a_2b_1}{b_1 - a_1} \right]$$

Perhatikan hasil kali gradien garis g dan h adalah:

$$\frac{-(b_1 - a_1)}{(b_2 - a_2)} \times \frac{(b_2 - a_2)}{(b_1 - a_1)} = -1.$$

Artinya kedua garis tegak lurus. **Contoh 3: Rumus Heron**

Rumus Heron menyatakan bahwa luas segitiga dengan panjang sisi-sisi a, b dan c adalah:

$$L = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{dengan } s = (a+b+c)/2,$$

atau bisa ditulis dalam bentuk lain:

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

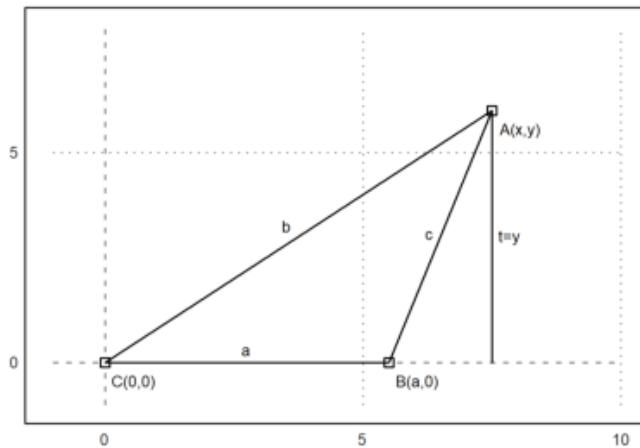
Untuk membuktikan hal ini kita misalkan C(0,0), B(a,0) dan A(x,y), b=AC, c=AB. Luas segitiga ABC adalah

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

Nilai y didapat dengan menyelesaikan sistem persamaan:

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

```
>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])
```

$$\begin{aligned} & -c^2 + b^2 + a^2 \\ & [x = \frac{2a^2}{4}, y = \frac{2a^2}{2}] \end{aligned}$$

```

sqrt(- c + 2 b c + 2 a c - b + 2 a b - a )
- -----] ,
2 a
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

```

Ekstrak solusi y

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

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

Kita mendapatkan formula Heron.

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

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

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

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

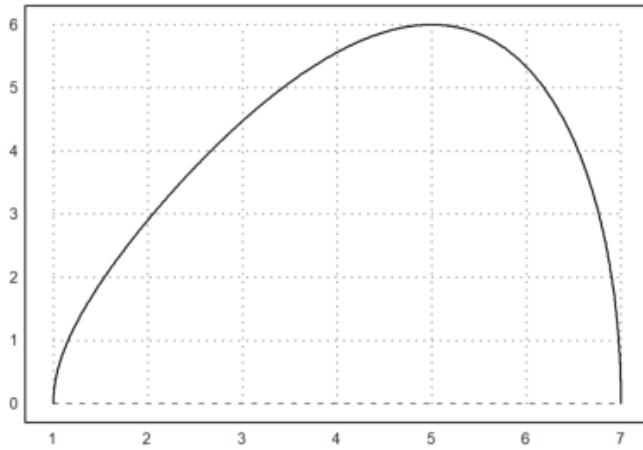
Tentu saja, setiap segitiga persegi panjang adalah kasus yang terkenal.

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

6

Dan jelas juga, bahwa ini adalah segitiga dengan luas maksimal dan kedua sisinya 3 dan 4.

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



The general case works too.

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

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

Now let us find the set of all points where $b+c=d$ for some constant d . It is well known that this is an ellipse.

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

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

Kasus umum juga berfungsi.

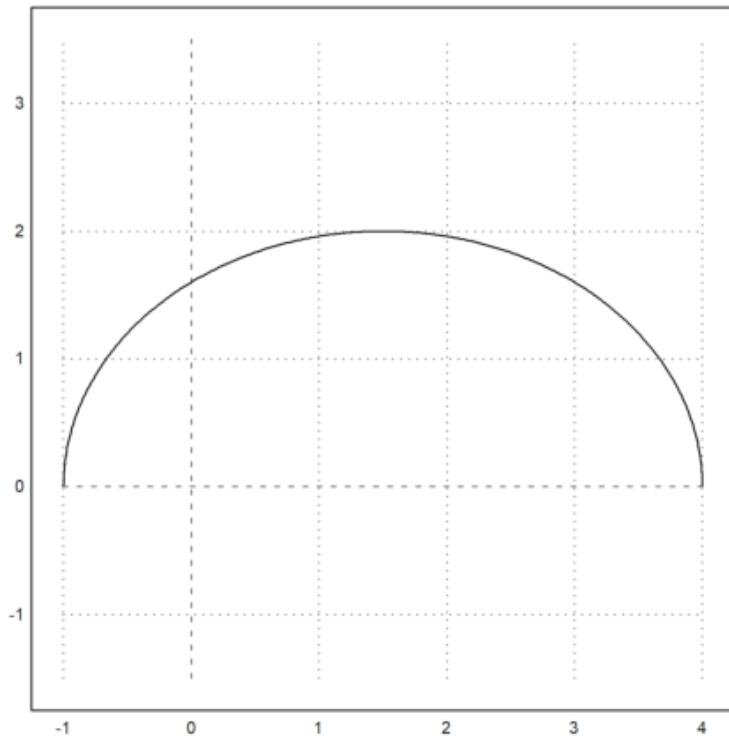
```
>function fx(a,c,d) &= rhs(s1[1]); $fx(a,c,d), function fy(a,c,d) &= rhs(s1[2]); $fy(a,c,d)
```

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



Sekarang kita bisa menggambar setnya. Sisi b bervariasi dari 1 hingga 4. Diketahui bahwa kita mendapatkan ellips.

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



Kita dapat memeriksa persamaan umum elips ini, yaitu.

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

di mana (x_m, y_m) 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 $x = 0$. Jadi luas segitiga dengan $a + b + c = d$ adalah maksimal, jika sama sisi. Kami ingin mendapatkan ini secara analitis.

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

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

Kami 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, memaksimalkan $H(a,b,c)^2$ terhadap $a+b+d=d$.

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

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

Kita bisa membuat plot situasinya.

Pertama, atur poin di Maxima

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

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

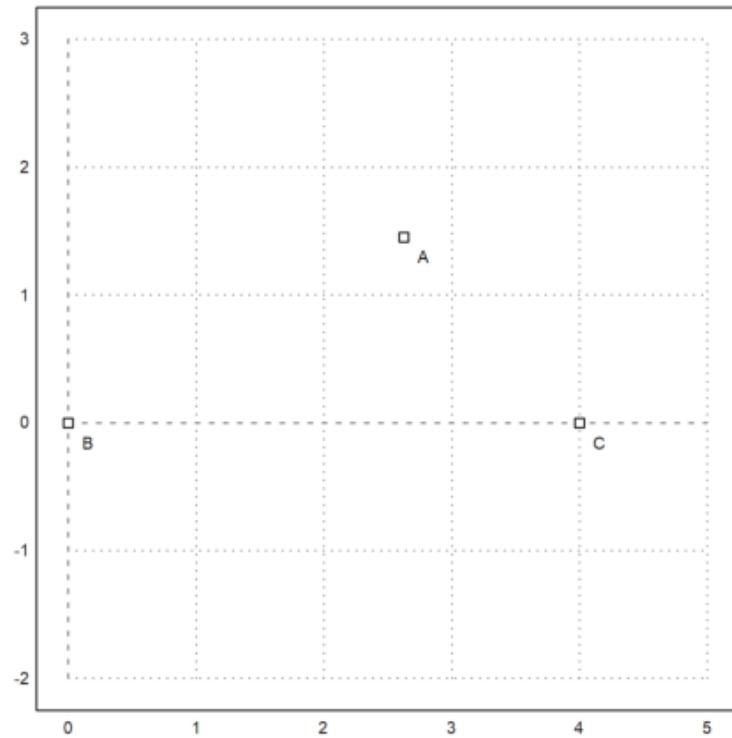
```
>B &= [0,0]; $B, C &= [a,0]; $C
```

$$[a, 0]$$

images/Shalih Abdillah_22305144009_ApKom6-046-lar

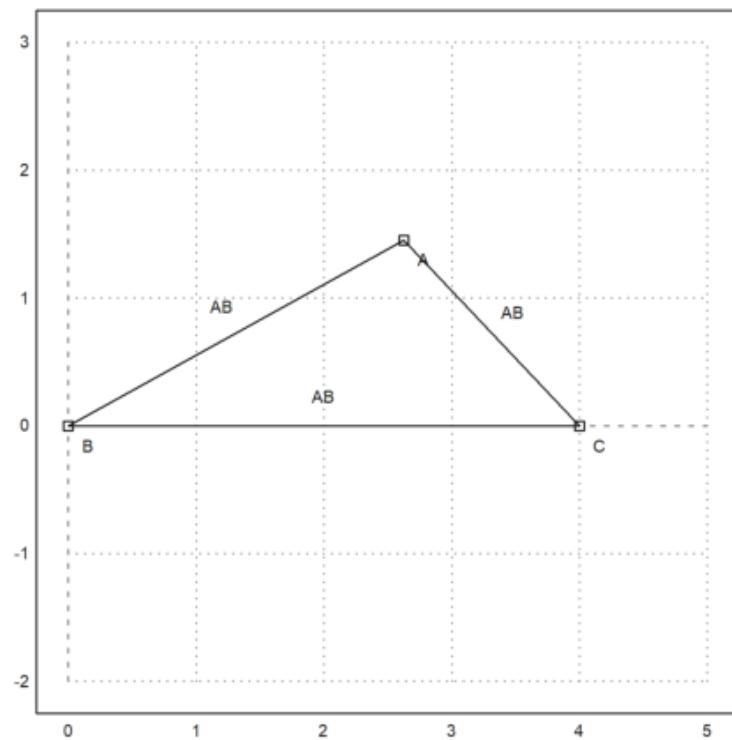
Kemudian atur rentang plot, dan plot poinnya.

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



Plot segmennya.

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



Hitung tengah tegak lurus di Maxima.

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

Dan bagian tengah dari keliling.

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

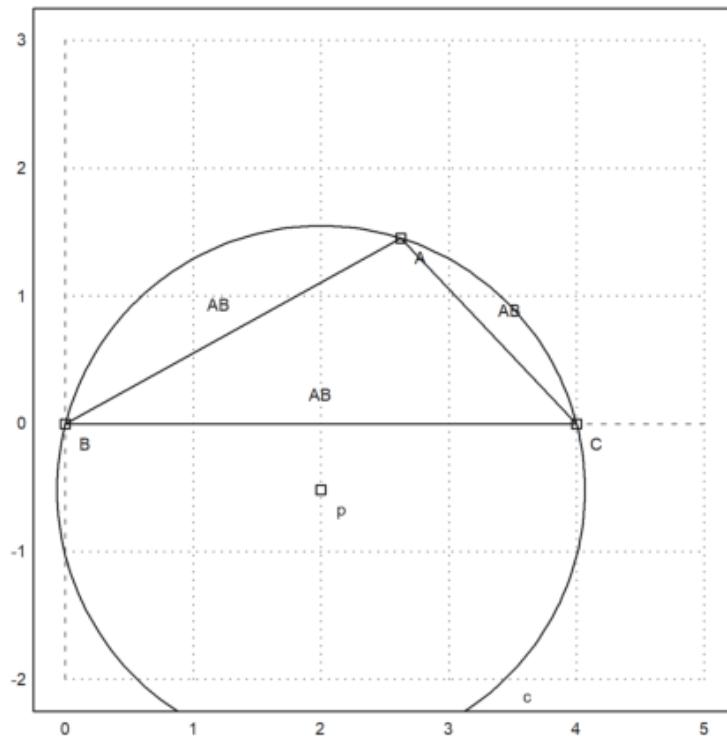
Kita mendapatkan rumus untuk jari-jari lingkaran.

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

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

Mari kita tambahkan ini ke plot.

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



Menggunakan geometri, kita mendapatkan rumus sederhana

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

untuk radius. Kita dapat memeriksa, apakah ini benar dengan Maxima. Maxima akan menfaktorkannya hanya jika kita mengkuadratkannya.

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

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

Contoh 4: Garis Euler dan Parabola

Garis euler adalah garis yang ditentukan dari segitiga yang tidak sama sisi. Ini adalah garis tengah segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga, termasuk pusat ortosentrum, sirkumenter, pusat massa, titik Exeter, dan pusat lingkaran sembilan titik segitiga.

Untuk demonstrasi, kami menghitung dan memplot garis Euler dalam segitiga.

Pertama, kami menentukan sudut segitiga di Euler. Kami menggunakan definisi, yang terlihat dalam ekspresi simbolik.

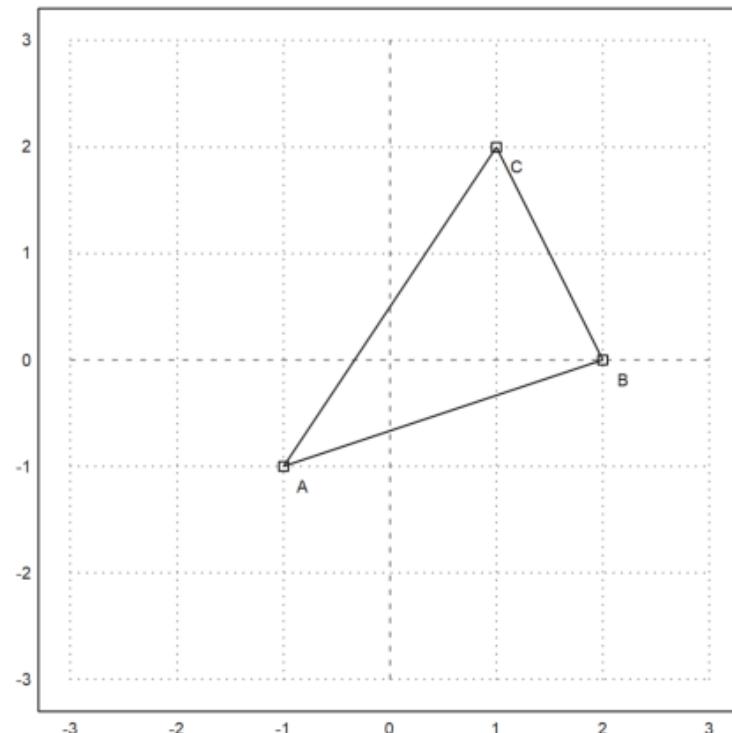
```
>A:=[-1,-1]; B:=[2,0]; C:=[1,2];
```

Untuk memplot objek geometris, kami menyiapkan area plot, dan menambahkan poin ke dalamnya. Semua plot objek geometris ditambahkan ke plot saat ini.

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

Kita juga bisa menambahkan sisi segitiga.

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



Berikut adalah luas segitiga menggunakan rumus 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 dapatkan rumus untuk baris ini.

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

$$3y - x = -2$$

Untuk bentuk Hesse, kita perlu menentukan titik, sehingga titik tersebut berada di sisi positif dari bentuk Hesse. Memasukkan titik menghasilkan jarak positif ke garis.

```
>$getHesseForm(c,x,y,C), $at(%,[x=C[1],y=C[2]])
```

$$\frac{7}{\sqrt{10}}$$

[\[images/Shalih Abdillah_22305144009_ApKom6-057-large\]\(https://drive.google.com/file/d/1JLWzXQDwvIjyfCgkVYUOOGdPmZGKoMn/view?usp=sharing\)](#)

Sekarang kita menghitung sirkit ABC.

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

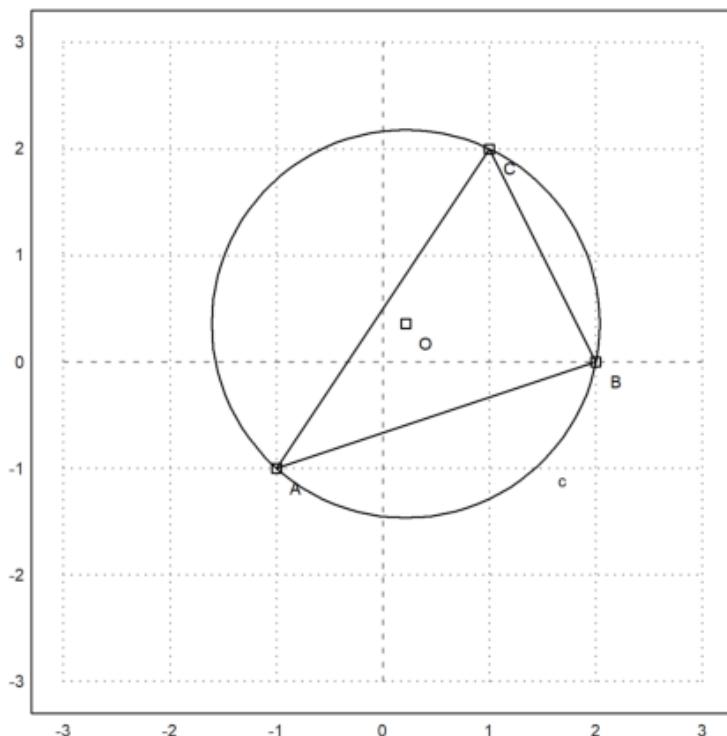
$$\left(y - \frac{5}{14}\right)^2 + \left(x - \frac{3}{14}\right)^2 = \frac{325}{98}$$

```
>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"):
```



Kita dapat menghitung perpotongan ketinggian di ABC (orthocenter) secara numerik dengan perintah berikut.

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

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

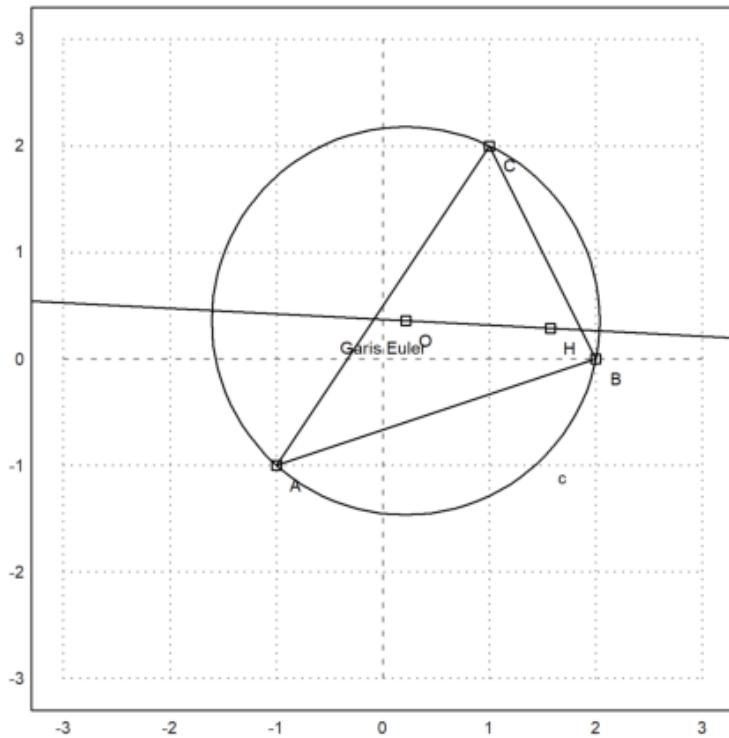
Sekarang kita dapat menghitung garis Euler dari segitiga tersebut.

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

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

Tambahkan ke plot kita.

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

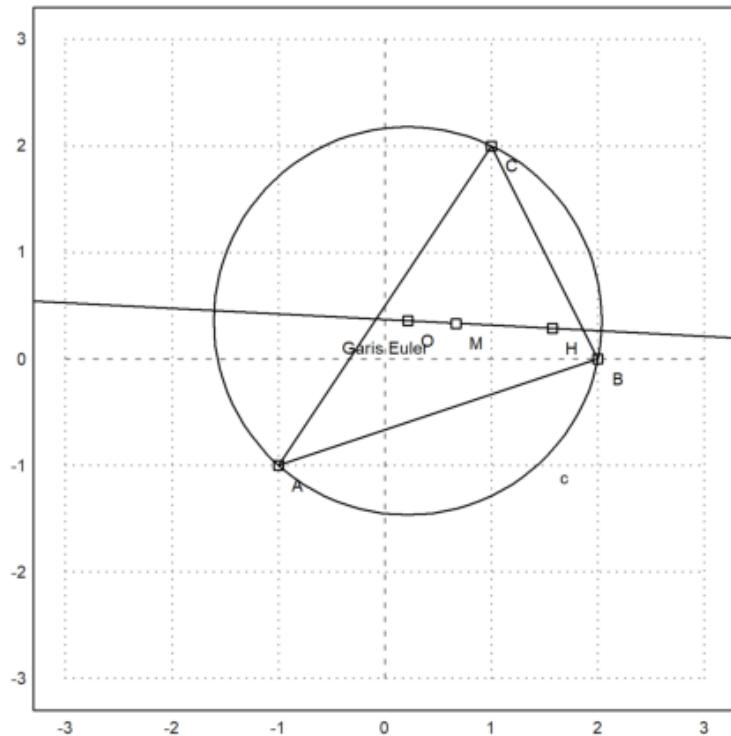


Pusat gravitasi harus berada di garis ini.

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

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

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



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

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

2

Fungsinya termasuk fungsi untuk sudut juga.

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

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

$60^\circ 15' 18.43''$

Persamaan untuk pusat lingkaran tidak terlalu bagus.

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

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

Mari kita hitung juga ekspresi jari-jari lingkaran yang tertulis.

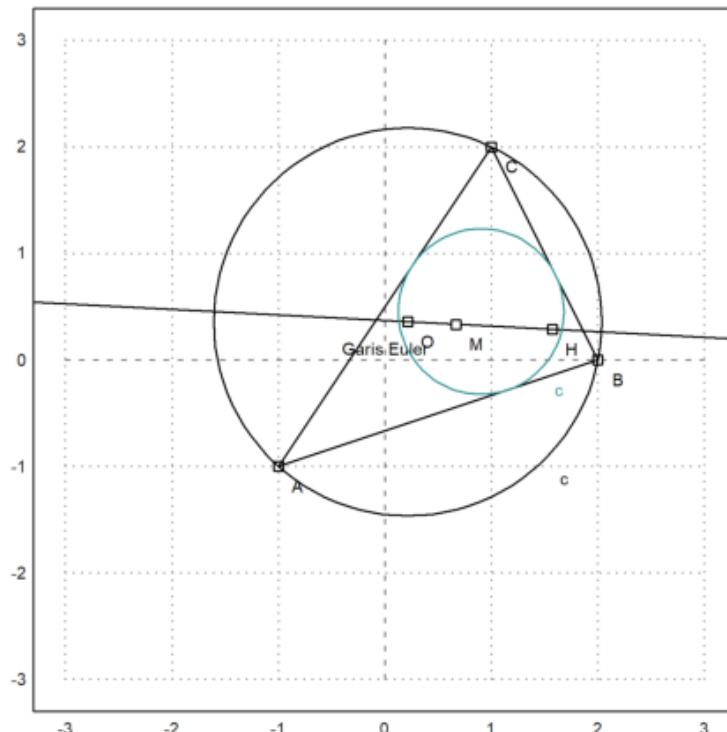
```
>r &= distance(Q, projectToLine(Q, lineThrough(A, B))) | ratsimp; $r
```

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

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

Mari kita tambahkan ini ke plot.

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



Parabola

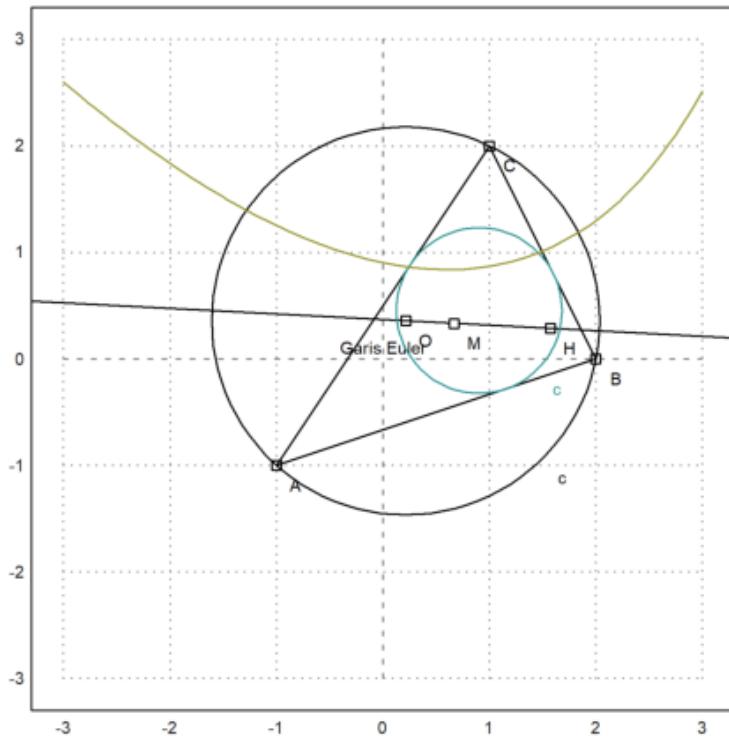
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

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

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

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

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



Ini seharusnya menjadi beberapa fungsi, tetapi pemecah default Maxima dapat menemukan solusi hanya, jika persamaan kita kuadratkan. Akibatnya, kami mendapatkan solusi palsu.

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

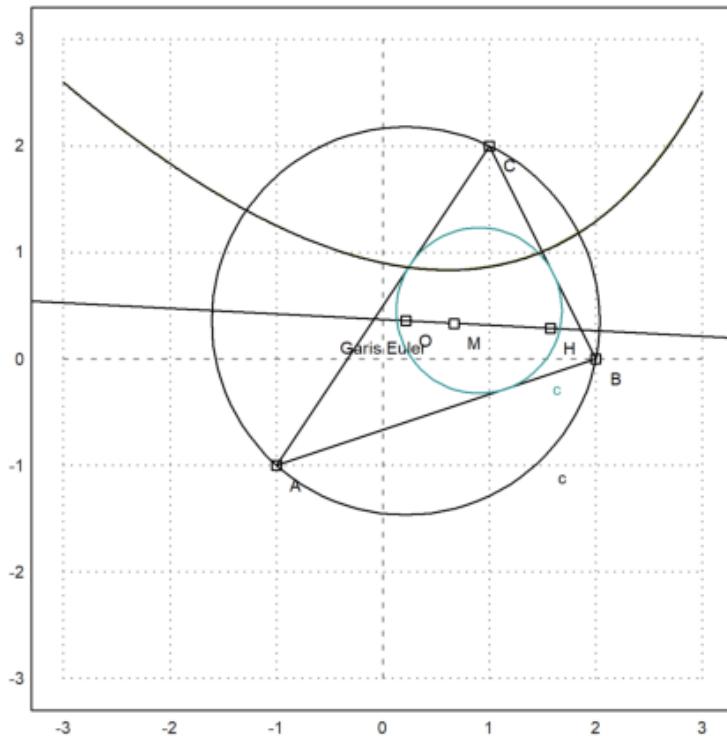
$$\begin{aligned}y &= -3x - \sqrt{70} \sqrt{9 - 2x} + 26, \\y &= -3x + \sqrt{70} \sqrt{9 - 2x} + 26\end{aligned}$$

Solusi pertama adalah

akar₁

Menambahkan solusi pertama ke pertunjukkan plot, bahwa itu memang jalan yang kita cari. Teori mengatakan kepada kita bahwa itu adalah parabola yang diputar.

```
>plot2d(&rhs(akar[1]),add=1):
```



```
>function g(x) &= rhs(akar[1]); $'g(x)= g(x)// fungsi yang mendefinisikan kurva di atas
```

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

```
>T &=[-1, g(-1)]; // ambil sebarang titik pada kurva tersebut
>dTC &= distance(T,C); $fullratsimp(dTC), $float(%) // jarak T ke C
```

$$2.135605779339061$$

images/Shalih Abdillah_22305144009_ApKom6-077-lar

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

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

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

$$2.135605779339061$$

images/Shalih Abdillah_22305144009_ApKom6-080-lar

Ternyata jarak T ke C sama dengan jarak T ke AB. Coba Anda pilih titik T yang lain dan ulangi perhitungan-perhitungan di atas untuk menunjukkan bahwa hasilnya juga sama.

Contoh 5: Trigonometri Rasional

Ini terinspirasi oleh ceramah N.J.Wildberger. Dalam bukunya "Proporsi Agung", Wildberger mengusulkan untuk menggantikan pengertian klasik tentang jarak dan sudut dengan kuadransi dan penyebaran. Dengan menggunakan ini, memang mungkin untuk menghindari fungsi trigonometri dalam banyak contoh, dan tetap "rasional".

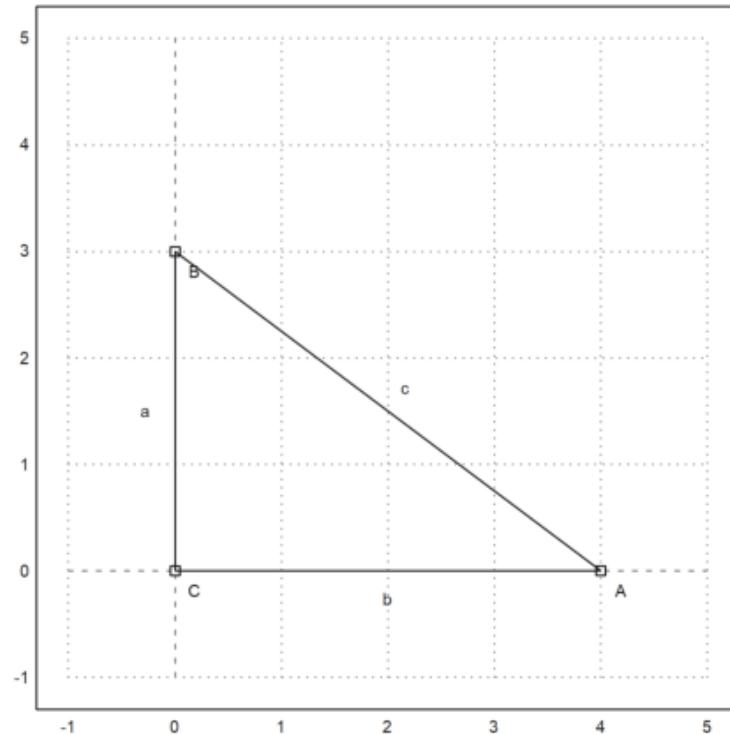
Berikut ini, saya memperkenalkan konsep, dan memecahkan beberapa masalah. Saya menggunakan perhitungan simbolik Maxima di sini, yang menyembunyikan keuntungan utama dari trigonometri rasional bahwa perhitungan dapat dilakukan dengan kertas dan pensil saja. Anda diundang untuk memeriksa hasil tanpa komputer.

Intinya adalah bahwa perhitungan rasional simbolis sering kali menghasilkan hasil yang sederhana. Sebaliknya, trigonometri klasik menghasilkan hasil trigonometri yang rumit, yang mengevaluasi ke pendekatan numerik saja.

```
>load geometry;
```

Untuk pendahuluan pertama, kami menggunakan segitiga persegi panjang dengan proporsi Mesir terkenal 3, 4 dan 5. Perintah berikut adalah perintah Euler untuk memplot geometri bidang yang terdapat dalam file Euler "geometry.e".

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



Tentu saja,

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

di mana wa adalah sudut di A. Cara biasa untuk menghitung sudut ini, adalah dengan melakukan invers dari fungsi sinus. Hasilnya adalah sudut yang tidak dapat dicerna, yang hanya dapat dicetak secara perkiraan.

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

36°52'11.63''

Trigonometri rasional mencoba menghindari hal ini.

Pengertian pertama dari trigonometri rasional adalah kuadran, yang menggantikan jarak. Faktanya, itu hanya kuadrat jarak. Berikut ini, a, b, dan c menunjukkan kuadran sisi-sisinya.

Teorema Pythagoras menjadi $a+b=c$ lalu.

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

$$25 = 25$$

Gagasan kedua dari trigonometri rasional adalah penyebarannya. Spread mengukur bukaan antar baris. Ini adalah 0, jika garis sejajar, dan 1, jika garis persegi panjang. Ini adalah kuadran 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 kuadrat dari segitiga persegi panjang mana pun dengan satu sudut di A.

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

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

Ini lebih mudah dihitung daripada sudut, tentu saja. Tetapi Anda kehilangan properti yang sudut dapat ditambahkan dengan mudah.

Tentu saja, kita dapat mengubah nilai perkiraan sudut wa menjadi sprad, dan mencetaknya sebagai pecahan.

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

9/25

Hukum cosinus dari trigonometri klasik diterjemahkan menjadi "hukum silang" berikut.

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

Di sini a, b, dan c adalah kuadran dari sisi-sisi segitiga, dan sa adalah sebaran di sudut A. Sisi a, seperti biasa, berlawanan dengan sudut A.

Hukum ini diimplementasikan dalam file geometry.e yang kami muat ke Euler.

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

$$\left[\left(bb - aa + \frac{7}{6} \right)^2, \left(bb - aa + \frac{7}{6} \right)^2, \left(bb - aa + \frac{5}{3\sqrt{2}} \right)^2 \right] = \left[\frac{14 bb (1 - saa)}{3}, \frac{14 bb (1 - saa)}{3}, \frac{5 2^{\frac{3}{2}} bb (1 - saa)}{3} \right]$$

Dalam kasus kami, kita mendapatkan

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

$$1024 = 1024$$

Mari kita gunakan crosslaw ini untuk mencari sebaran di A. Untuk melakukan ini, kita menghasilkan crosslaw untuk kuadran a, b, dan c, dan menyelesaiakannya untuk sebaran yang tidak diketahui sa. Anda dapat melakukan ini dengan tangan dengan mudah, tetapi saya menggunakan Maxima. Tentu saja, kami mendapatkan hasilnya, kami sudah mendapatkannya.

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

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

 images/Shalih Abdillah_22305144009_ApKom6-089-large

Kami sudah tahu ini. Definisi penyebaran adalah kasus khusus dari hukum lintas hukum.

Kita juga bisa menyelesaikan ini untuk umum a, b, c. Hasilnya adalah rumus yang menghitung sebaran sudut segitiga berdasarkan kuadran ketiga sisinya.

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

$$\left[\left[\frac{168 bb x + 36 bb^2 + (-72 aa - 84) bb + 36 aa^2 - 84 aa + 49}{36}, \frac{168 bb x + 36 bb^2 + (-72 aa - 84) bb + 36 aa^2 - 84 aa + 49}{36} \right], \dots \right]$$

Kita bisa membuat fungsi dari hasilnya. Fungsi seperti itu sudah ditentukan dalam file geometry.e Euler.

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

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

Sebagai contoh, kita bisa menggunakananya untuk menghitung sudut segitiga bersisi

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

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

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

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

Ini adalah sudut dalam derajat.

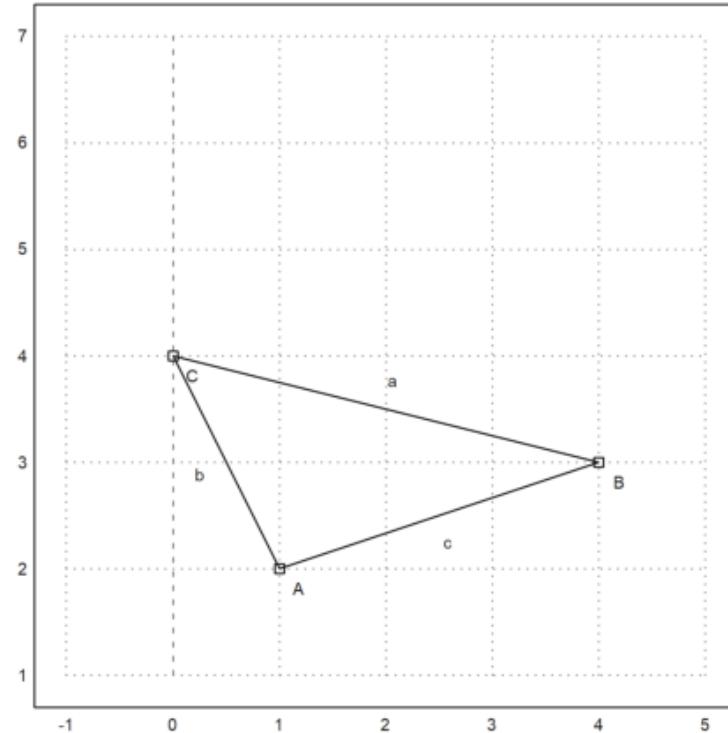
```
>degsprint(arcsin(sqrt(6/7)))
```

$67^\circ 47' 32.44''$

Contoh lain

Sekarang, mari kita coba contoh yang lebih canggih.
Kami mengatur 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;
```



Menggunakan Pythagoras, mudah untuk menghitung jarak antara dua titik. Saya pertama kali menggunakan jarak fungsi file Euler untuk geometri. Jarak fungsi menggunakan geometri klasik.

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

$$\sqrt{10}$$

Euler juga memiliki fungsi kuadrans antara dua titik.

Dalam contoh berikut, karena $c + b$ bukan a , segitiga tidak persegi panjang.

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

17

images/Shalih Abdillah_22305144009_ApKom6-097-lar
images/Shalih Abdillah_22305144009_ApKom6-098-lar

Pertama, mari kita hitung sudut tradisional. Fungsi computeAngle menggunakan metode biasa berdasarkan perkalian titik dari dua vektor. Hasilnya adalah beberapa pendekatan floating point.

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

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

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

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

```
>wb &= computeAngle(A,B,C); $wb, $(wb/pi*180)()
```

$$\arccos \left(\frac{11}{\sqrt{10} \sqrt{17}} \right)$$

32.4711922908

Menggunakan pensil dan kertas, kita bisa melakukan hal yang sama dengan hukum silang. Kami memasukkan kuadran a , b , dan c ke dalam hukum silang dan menyelesaikan untuk x .

```
>$crosslaw(a,b,c,x), $solve(% ,x), // (b+c-a)^=4b.c(1-x)
```

$$\left[x = \frac{49}{50} \right]$$



images/Shalih Abdillah_22305144009_ApKom6-105-large

Artinya, fungsi penyebaran 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. Itu menyelesaikan istilah $\sin(\arccos(...))$ menjadi hasil pecahan. Kebanyakan siswa tidak dapat melakukan ini.

```
>$sin(computeAngle(A,B,C))^2
```

$$\frac{49}{170}$$

Setelah kita mendapatkan sebaran di B, kita bisa menghitung tinggi ha di sisi a. Ingat bahwa

$$s_b = \frac{h_a}{c}$$

Menurut definisi.

```
>ha &= c*sb; $ha
```

$$\frac{49}{17}$$

Gambar berikut telah diproduksi dengan program geometri C.a.R., yang dapat menggambar kuadran dan menyebar.

gambar : (20) Rational_Geometry_CaR.png

Menurut definisi, panjang ha adalah akar kuadrat dari kuadrannya.

```
>$sqrt(ha)
```

$$\frac{7}{\sqrt{17}}$$

Sekarang kita bisa menghitung luas segitiga. Jangan lupa, bahwa kita berurusan dengan kuadran!

```
>$sqrt(ha)*sqrt(a)/2
```

$$\frac{7}{2}$$

Rumus determinan yang biasa menghasilkan hasil yang sama.

```
>$areaTriangle(B,A,C)
```

$$\frac{7}{2}$$

Formula Heron

Sekarang, mari kita selesaikan masalah ini secara umum!

```
>&remvalue(a,b,c,sb,ha);
```

Pertama-tama kita menghitung spread di B untuk segitiga dengan sisi a, b, dan c. Kemudian kami menghitung luas area yang dikuadratkan ("kuadrea"?), Memfaktorkannya dengan Maxima, dan kami mendapatkan rumus Heron yang terkenal.

Memang, ini sulit dilakukan dengan pensil dan kertas.

```
>$spread(b^2,c^2,a^2), $factor(%*c^2*a^2/4)
```

$$\frac{(-c + b + a) (c - b + a) (c + b - a) (c + b + a)}{16}$$

[images/Shalih Abdillah_22305144009_ApKom6-114-larg

Aturan Triple Spread

Kerugian dari spread adalah bahwa mereka tidak lagi hanya menambahkan sudut serupa. Namun, tiga sebaran segitiga memenuhi aturan "penyebaran rangkap tiga" berikut.

```
>&remvalue(sa,sb,sc); $triplespread(sa,sb,sc)
```

$$(sc + sb + sa)^2 = 2 (sc^2 + sb^2 + sa^2) + 4 sa sb sc$$

Aturan ini berlaku untuk tiga sudut yang bertambah menjadi 180° .

$$\alpha + \beta + \gamma = \pi$$

Sejak penyebaran

$$\alpha, \pi - \alpha$$

sama, aturan penyebaran tiga kali lipat juga benar, jika

$$\alpha + \beta = \gamma$$

Karena penyebaran sudut negatif adalah sama, aturan penyebaran tiga kali lipat juga berlaku, jika

$$\alpha + \beta + \gamma = 0$$

Misalnya, kita dapat menghitung sebaran sudut 60° . Ini 3/4. Persamaan memiliki solusi kedua, di mana semua spread adalah 0.

```
>$solve(triplespread(x,x,x),x)
```

$$\left[x = \frac{3}{4}, x = 0 \right]$$

Sebaran 90° jelaslah 1. Jika dua sudut dijumlahkan menjadi 90° , penyebarannya menyelesaikan persamaan penyebaran rangkap tiga dengan a, b, 1. Dengan perhitungan berikut kita mendapatkan $a+b=1$.

```
>$triplespread(x,y,1), $solve(% ,x)
```

$$[x = 1 - y]$$

[images/Shalih Abdillah_22305144009_ApKom6-122-larg

Karena penyebaran $180^\circ - t$ sama dengan penyebaran t , rumus penyebaran rangkap tiga juga berlaku, jika satu sudut adalah jumlah atau perbedaan dari dua sudut lainnya.

Jadi kita bisa menemukan sebaran sudut berlipat ganda. Perhatikan bahwa ada dua solusi lagi. Kami menjadikannya sebuah fungsi.

```
> $solve(triplespread(a,a,x),x), function doublespread(a) &= factor(rhs(%[1]))
```

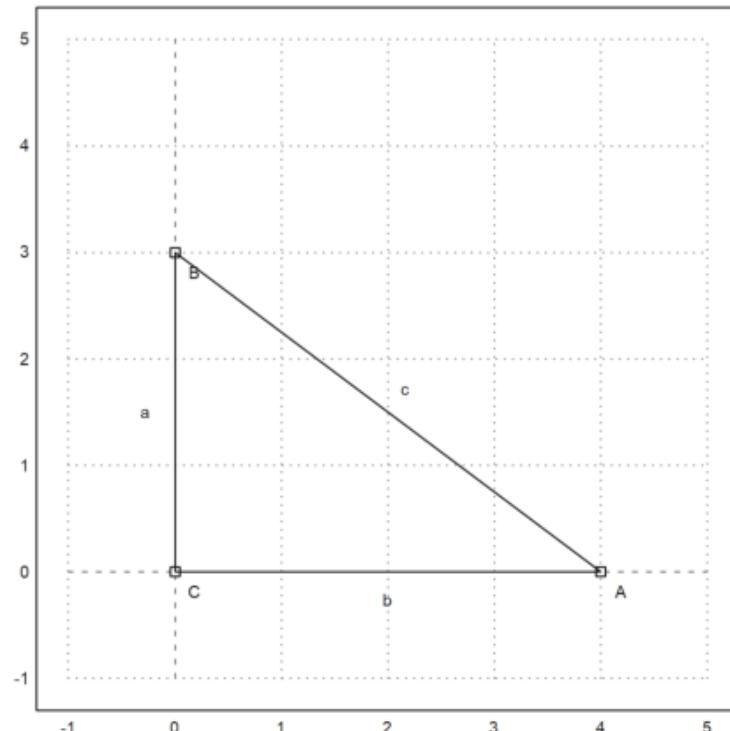
$$[x = 4a - 4a^2, x = 0]$$

$$- 4(a - 1)a$$

Pembagi Sudut

Ini situasinya, kita sudah tahu.

```
> 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 bisektor sudut pada A. Tapi kita ingin menyelesaiakannya untuk umum a, b, c.

```
> &remvalue(a,b,c);
```

Jadi pertama-tama kita menghitung sebaran sudut terbagi di A, menggunakan rumus sebaran rangkap tiga. Masalah dengan rumus ini muncul lagi. Ini memiliki dua solusi. Kami harus memilih yang benar. Solusi lainnya mengacu pada sudut terbagi 180° -wa.

```
>$triplespread(x,x,a/(a+b)), $solve(% ,x), sa2 &= rhs(%[1]); $sa2
```

$$\frac{-\sqrt{b}\sqrt{b+a}+b+a}{2b+2a}$$



images/Shalih Abdillah_22305144009_ApKom6-126-large



images/Shalih Abdillah_22305144009_ApKom6-127-large

Mari kita periksa persegi panjang Mesir.

```
>$sa2 with [a=3^2,b=4^2]
```

$$\frac{1}{10}$$

Kami dapat mencetak sudut di Euler, setelah mentransfer penyebaran ke radian.

```
>wa2 := arcsin(sqrt(1/10)); degprint(wa2)
```

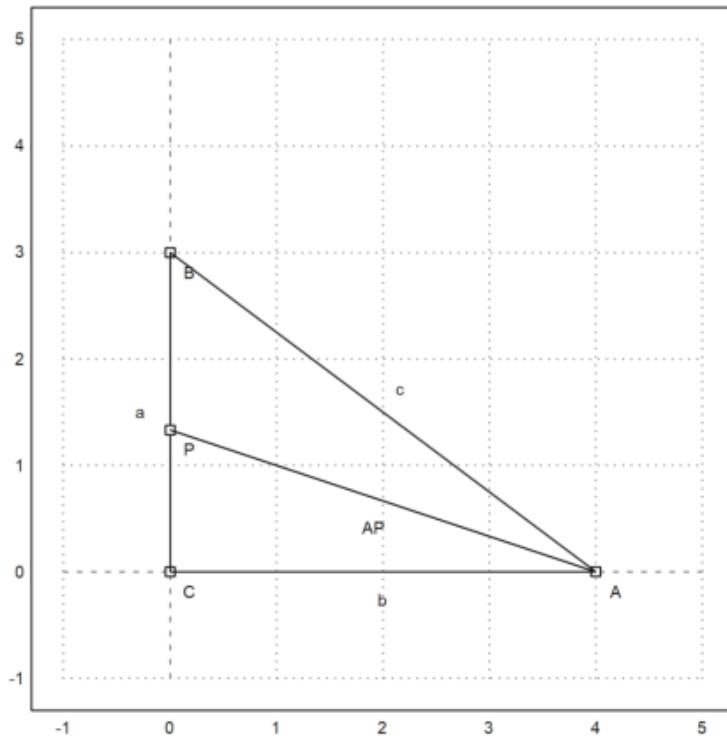
$18^\circ 26' 5.82''$

Titik P adalah perpotongan dari garis bagi sudut dengan sumbu y.

```
>P := [0,tan(wa2)*4]
```

[0, 1.33333]

```
>plotPoint(P,"P"); plotSegment(A,P):
```



Mari kita periksa sudut dalam contoh spesifik kita.

```
>computeAngle(C,A,P), computeAngle(P,A,B)
```

```
0.321750554397
0.321750554397
```

Sekarang kita menghitung panjang bisektor AP.

Kita menggunakan teorema sinus di segitiga APC. Teorema ini menyatakan bahwa

$$\frac{BC}{\sin(w_a)} = \frac{AC}{\sin(w_b)} = \frac{AB}{\sin(w_c)}$$

memegang di segitiga apa pun. Persegi itu, itu diterjemahkan ke dalam apa yang disebut "hukum penyebaran"

$$\frac{a}{s_a} = \frac{b}{s_b} = \frac{c}{s_b}$$

dimana a, b, c menunjukkan qudrance.

Karena BPA sebaran adalah $1-sa^2$, kita dapatkan darinya bisa $/ 1 = b / (1-sa^2)$ dan dapat menghitung bisa (kuadran garis-garis).

```
>&factor(ratsimp(b/(1-sa^2))); bisa &= %; $bisa
```

$$\frac{2b(b+a)}{\sqrt{b}\sqrt{b+a} + b+a}$$

Mari kita periksa rumus ini untuk nilai Mesir kita.

```
>sqrt(mxmeval("at(bisa,[a=3^2,b=4^2])")), distance(A,P)
```

4.21637021356

4.21637021356

Kami juga dapat menghitung P menggunakan rumus spread.

```
>py&=factor(ratsimp(sa2*bisa)); $py
```

$$-\frac{b \left(\sqrt{b} \sqrt{b+a}-b-a\right)}{\sqrt{b} \sqrt{b+a+b+a}}$$

Nilainya sama dengan yang kita dapatkan dengan rumus trigonometri.

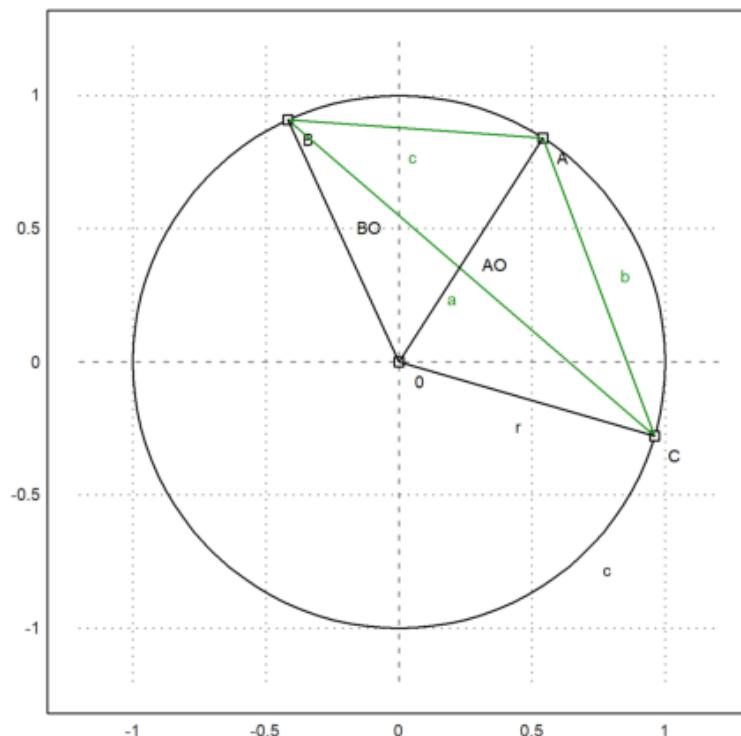
```
>sqrt(mxmeval("at(py,[a=3^2,b=4^2])"))
```

1.33333333333

Sudut Akord

Perhatikan situasi berikut.

```
>setPlotRange(1.2); ...
>color(1); plotCircle(circleWithCenter([0,0],1)); ...
>A:=[cos(1),sin(1)]; B:=[cos(2),sin(2)]; C:=[cos(6),sin(6)]; ...
>plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...
>color(3); plotSegment(A,B,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a"); ...
>color(1); O:=[0,0]; plotPoint(O,"O"); ...
>plotSegment(A,O); plotSegment(B,O); plotSegment(C,O,"r"); ...
>insimg;
```



Kita bisa menggunakan Maxima untuk menyelesaikan rumus sebaran rangkap tiga untuk sudut di pusat O untuk r. Jadi kita mendapatkan rumus untuk jari-jari kuadrat dari keliling dalam hal kuadrat sisi. Kali ini, Maxima menghasilkan beberapa angka nol yang kompleks, yang kita abaikan.

```
>&remvalue(a,b,c,r); // hapus nilai-nilai sebelumnya untuk perhitungan baru
>rabc &= rhs(solve(triplespread(spread(b,r,r),spread(a,r,r),spread(c,r,r)),r)[4]); $rabc
```

$$-\frac{abc}{c^2 - 2bc + a(-2c - 2b) + b^2 + a^2}$$

Kita bisa menjadikannya sebagai fungsi Euler.

```
>function periradius(a,b,c) &= rabc;
```

Mari kita periksa hasilnya untuk poin A, B, C kita.

```
>a:=quadrance(B,C); b:=quadrance(A,C); c:=quadrance(A,B);
```

Radiusnya memang 1.

```
>periradius(a,b,c)
```

1

Faktanya, penyebaran CBA hanya bergantung pada b dan c. Ini adalah teorema sudut akord.

```
>$spread(b,a,c)*rabc | ratsimp
```

$$\frac{b}{4}$$

Sebenarnya sebarannya adalah $b/(4r)$, dan kita melihat bahwa sudut akor b adalah setengah dari sudut tengah.

```
>$doublespread(b/(4*r))-spread(b,r,r) | ratsimp
```

0

Contoh 6: Jarak Minimal pada Bidang

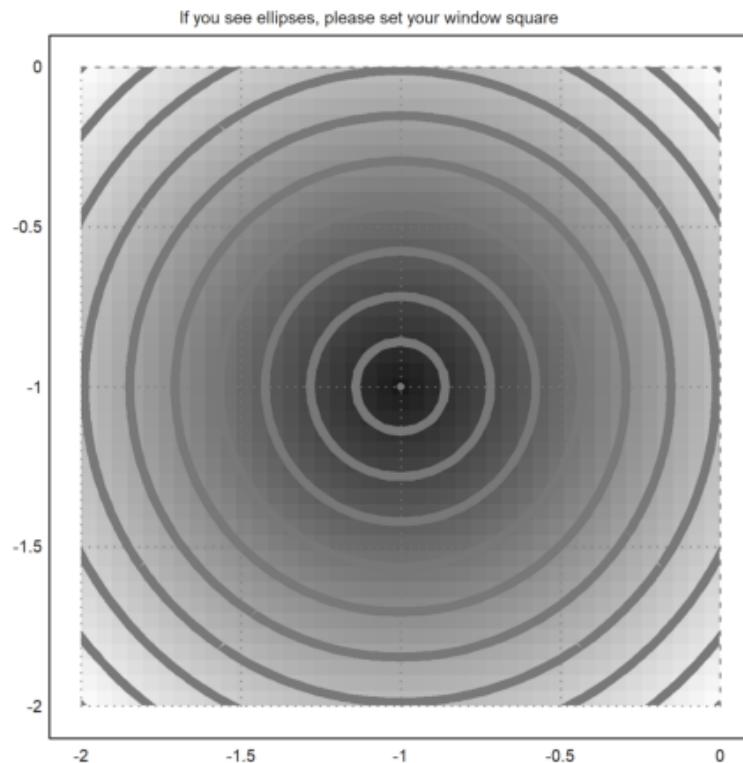
Catatan awal

Fungsi yang, ke titik M di bidang, menetapkan jarak AM antara titik tetap A dan M, memiliki garis level yang agak sederhana: lingkaran berpusat di A.

```

>&remvalue();
>A=[-1,-1];
>function d1(x,y):=sqrt((x-A[1])^2+(y-A[2])^2)
>fcontour("d1",xmin=-2,xmax=0,ymin=-2,ymax=0,hue=1, ...
>title="If you see ellipses, please set your window square":

```

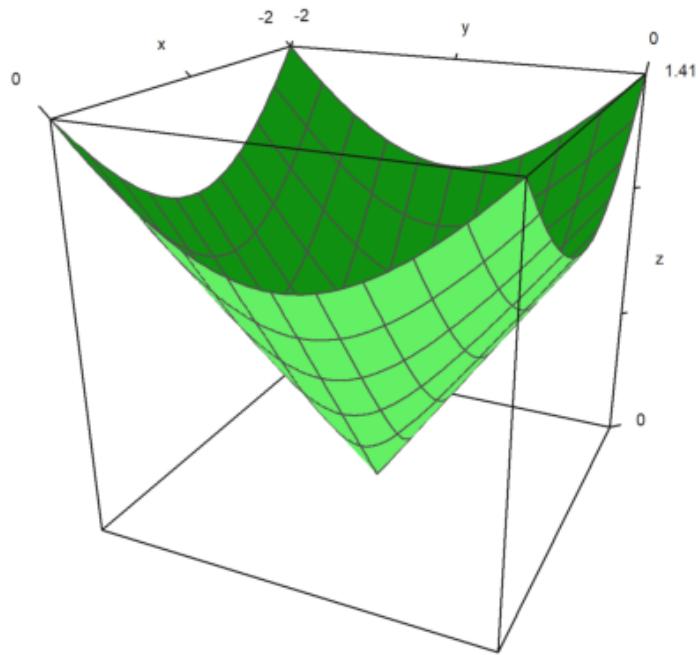


dan grafiknya juga agak sederhana: bagian atas kerucut:

```

>plot3d("d1",xmin=-2,xmax=0,ymin=-2,ymax=0):

```

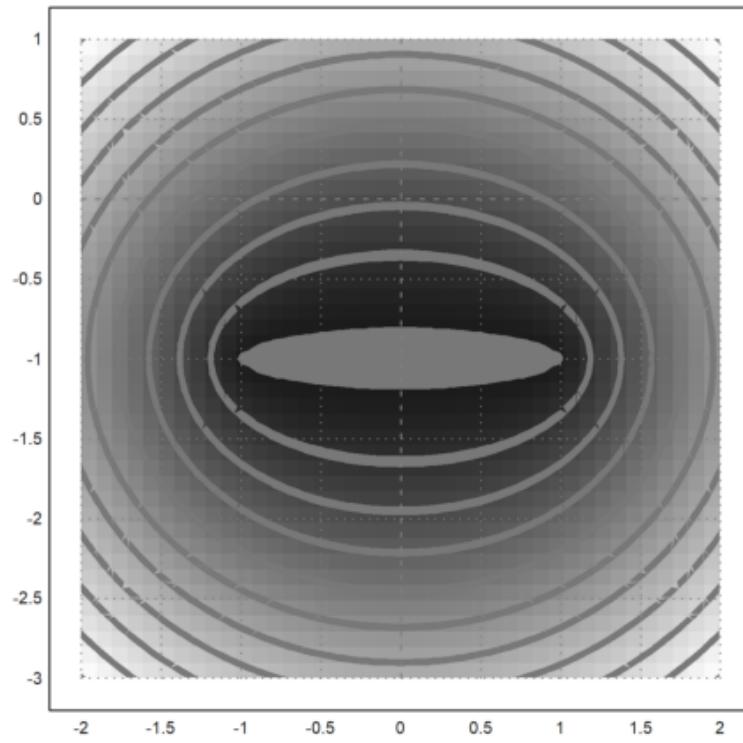


Tentu saja minimal 0 dicapai di A.

Dua titik

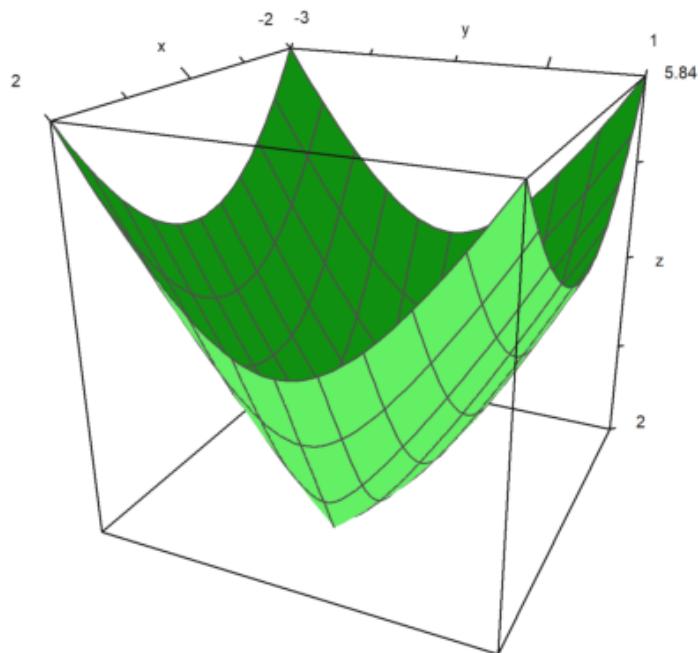
Sekarang kita melihat fungsi $MA + MB$ dimana A dan B adalah dua titik (tetap). Ini adalah "fakta yang terkenal" bahwa kurva level adalah elips, titik fokusnya adalah A dan B; kecuali untuk minimum AB yang konstan pada segmen [AB]:

```
>B=[1,-1];
>function d2(x,y):=d1(x,y)+sqrt((x-B[1])^2+(y-B[2])^2)
>fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
```



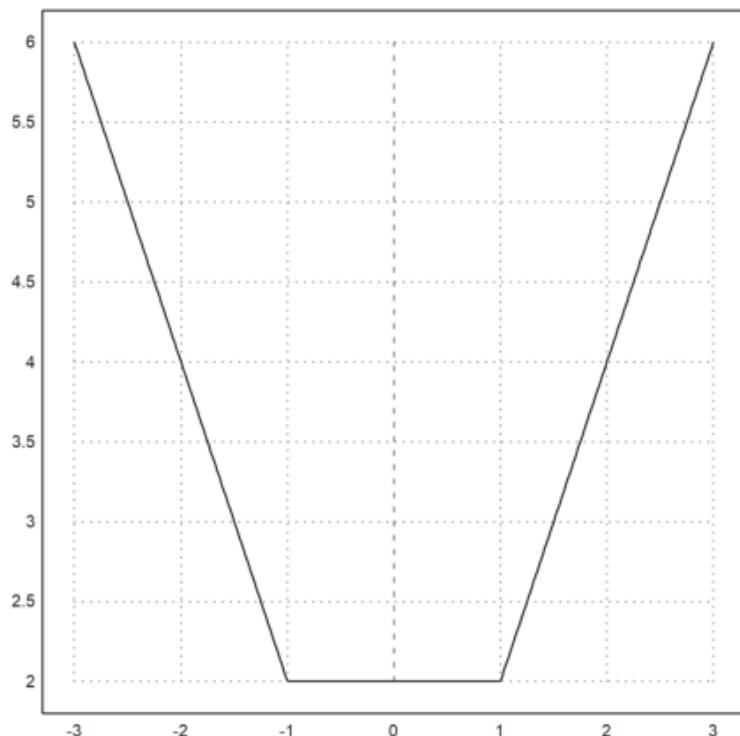
Grafiknya lebih menarik:

```
>plot3d("d2", xmin=-2, xmax=2, ymin=-3, ymax=1) :
```



Batasan ke baris (AB) lebih terkenal:

```
>plot2d("abs(x+1)+abs(x-1)",xmin=-3,xmax=3):
```



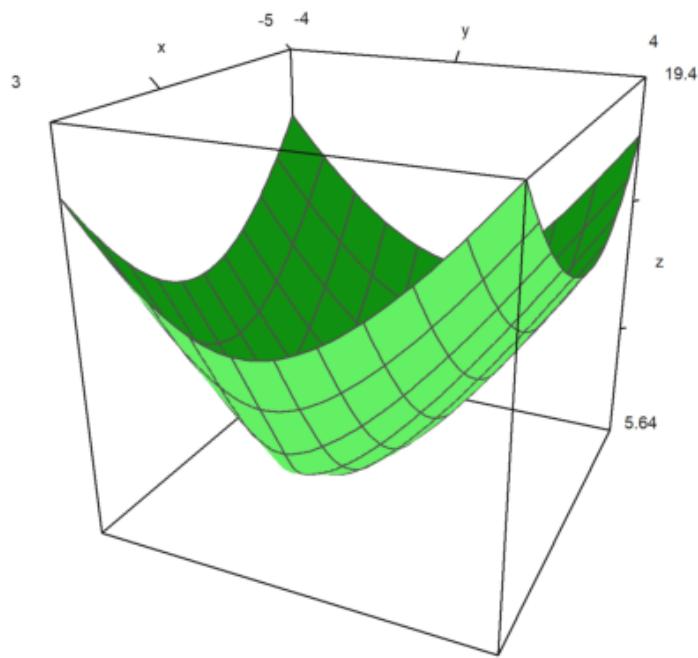
Tiga titik

Sekarang hal-hal menjadi kurang sederhana: Sedikit kurang diketahui bahwa $MA+MB+MC$ mencapai minimumnya pada satu titik bidang tetapi untuk menentukannya kurang sederhana:

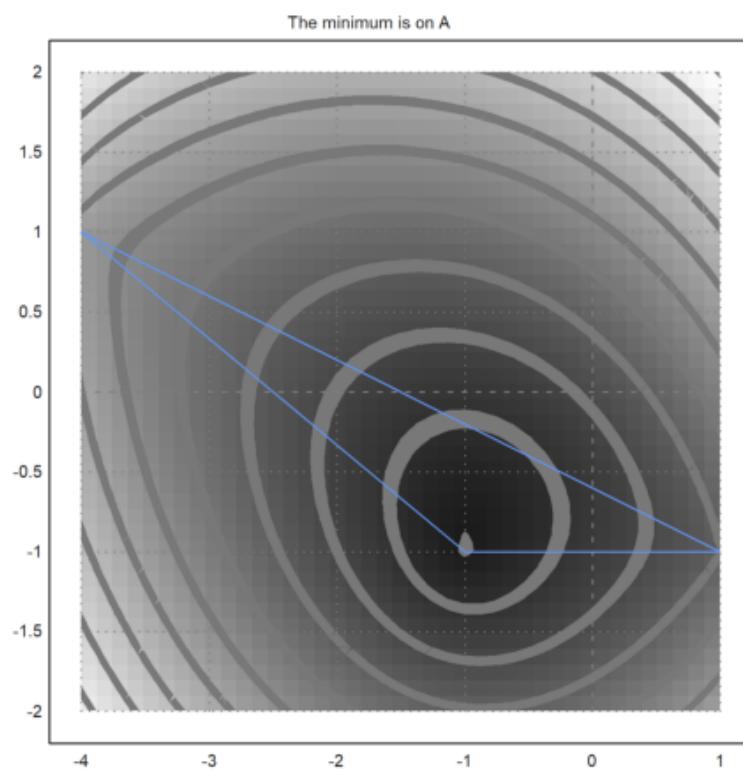
- 1) Jika salah satu sudut segitiga ABC lebih dari 120° (katakanlah dalam A), maka minimum tercapai pada titik ini (katakanlah AB+AC).

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;
```

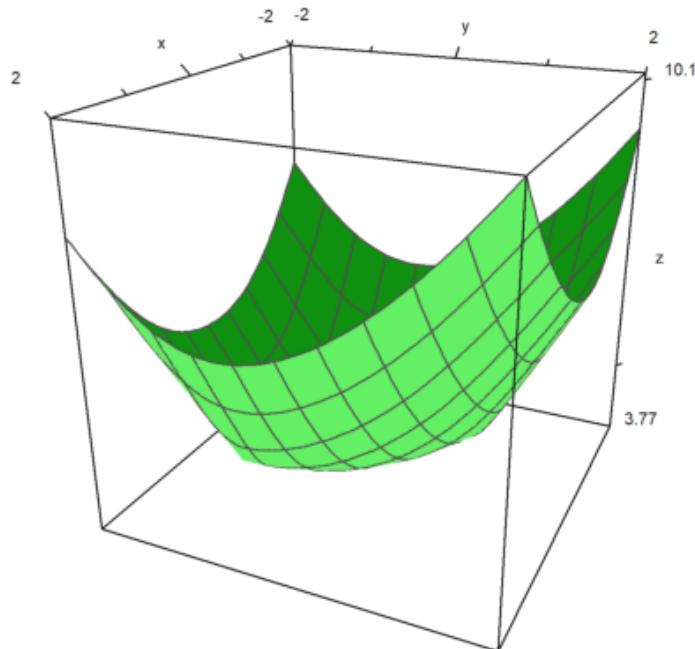


```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The minimum is on A");
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
>insimg;
```

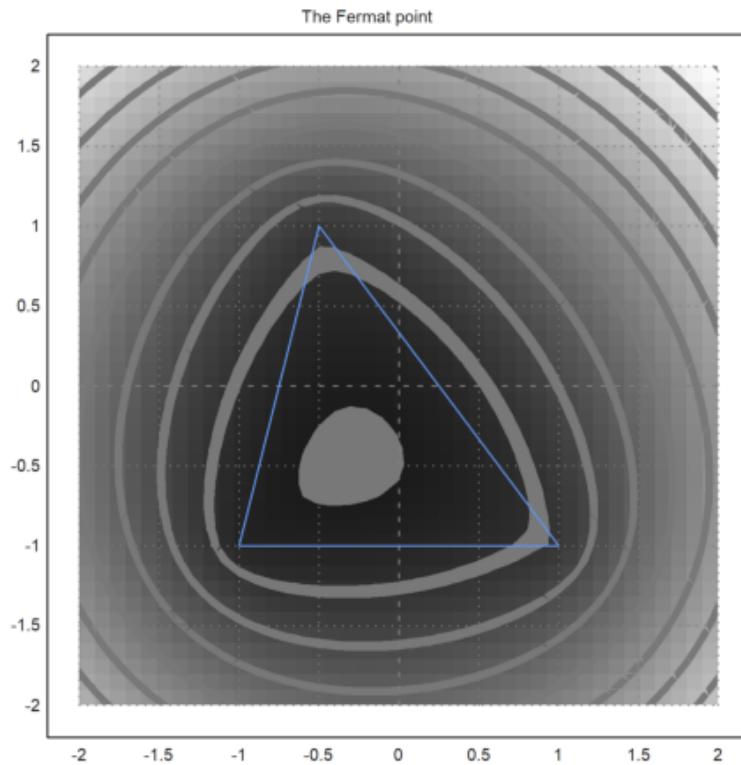


2) Tetapi jika semua sudut segitiga ABC kurang dari 120° , minimum berada pada titik F di bagian dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi ABC dengan sudut yang sama (lalu masing-masing 120°):

```
>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;
```



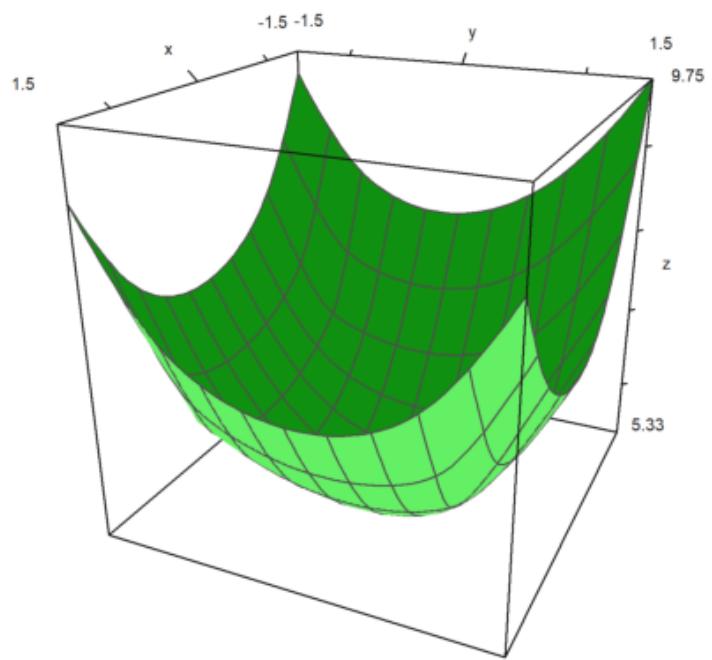
Merupakan kegiatan yang menarik untuk mewujudkan gambar di atas dengan perangkat lunak geometri; sebagai contoh, saya tahu soft tertulis di Java yang memiliki instruksi "garis kontur" ...

Semua ini di atas telah ditemukan oleh seorang hakim Prancis bernama Pierre de Fermat; dia menulis surat kepada para penggila lainnya seperti pendeta Marin Mersenne dan Blaise Pascal yang bekerja di bagian pajak penghasilan. Jadi titik unik F sehingga $FA + FB + FC$ minimal disebut titik Fermat segitiga. Tetapi tampaknya beberapa tahun sebelumnya, Torricelli Italia telah menemukan titik ini sebelum Fermat melakukannya! Pokoknya tradisinya adalah mencatat poin ini ...

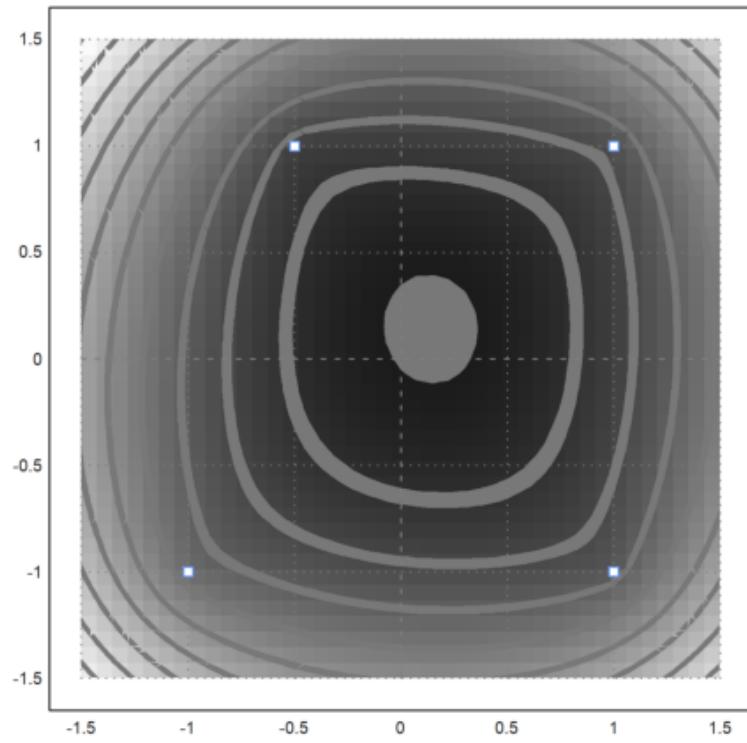
Empat titik

Langkah selanjutnya adalah menambahkan titik D ke-4 dan mencoba meminimalkan $MA + MB + MC + MD$; katakanlah bahwa Anda adalah operator TV kabel dan ingin mencari di bidang mana Anda harus meletakkan antena sehingga Anda dapat memberi makan empat desa dan menggunakan kabel sesedikit mungkin!

```
>D=[1,1];
>function d4(x,y):=d3(x,y)+sqrt((x-D[1])^2+(y-D[2])^2)
>plot3d("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5):
```



```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);  
>P=(A_B_C_D)'; plot2d(P[1],P[2],points=1,add=1,color=12);  
>insimg;
```



Masih ada minimum dan tidak ada yang dicapai pada simpul A, B, C atau D:

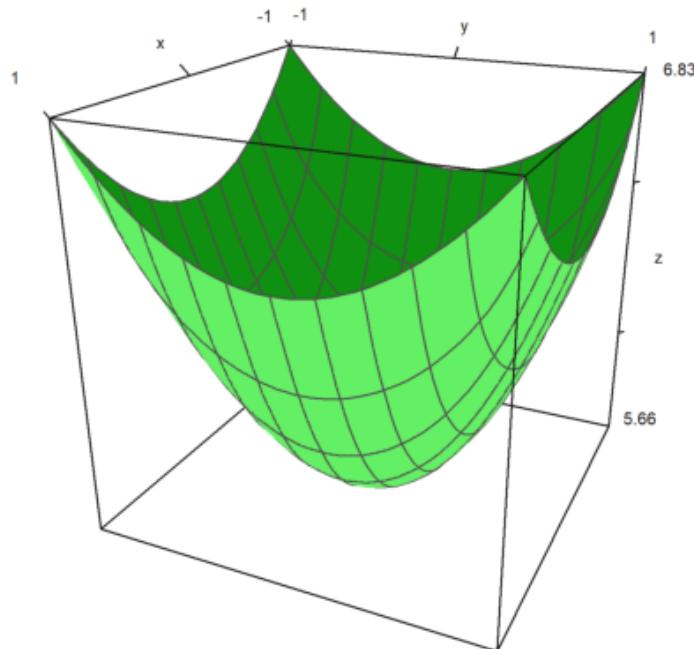
```
>function f(x):=d4(x[1],x[2])
>neldermin("f", [0.2,0.2])
```

```
[0.142858, 0.142857]
```

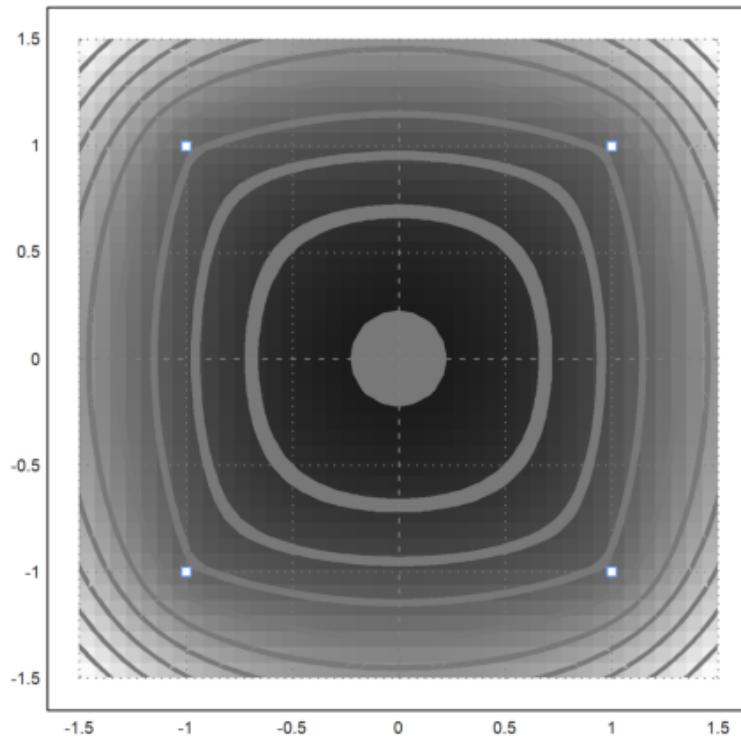
Tampaknya dalam kasus ini, koordinat titik optimal rasional atau mendekati rasional ...

Sekarang ABCD adalah bujur sangkar, kami berharap bahwa titik optimal adalah pusat ABCD:

```
>C=[-1,1];
>plot3d("d4",xmin=-1,xmax=1,ymin=-1,ymax=1):
```



```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
>P=(A_B_C_D)'; plot2d(P[1],P[2],add=1,color=12,points=1);
>insimg;
```



Contoh 7: Bola Dandelin dengan Povray

Anda dapat menjalankan demonstrasi ini, jika Anda memiliki Povray diinstal, dan pvcgine.exe di jalur program.

Pertama kami menghitung jari-jari bola.

Jika Anda melihat gambar di bawah, Anda melihat bahwa kita membutuhkan dua lingkaran yang menyentuh dua garis yang membentuk kerucut, dan satu garis yang membentuk bidang yang memotong kerucut.

Kami menggunakan file geometry.e dari Euler untuk ini.

```
>load geometry;
```

Pertama, dua garis yang membentuk kerucut.

```
>g1 &= lineThrough([0,0], [1,a])
```

```
[ - a, 1, 0 ]
```

```
>g2 &= lineThrough([0,0], [-1,a])
```

```
[ - a, - 1, 0 ]
```

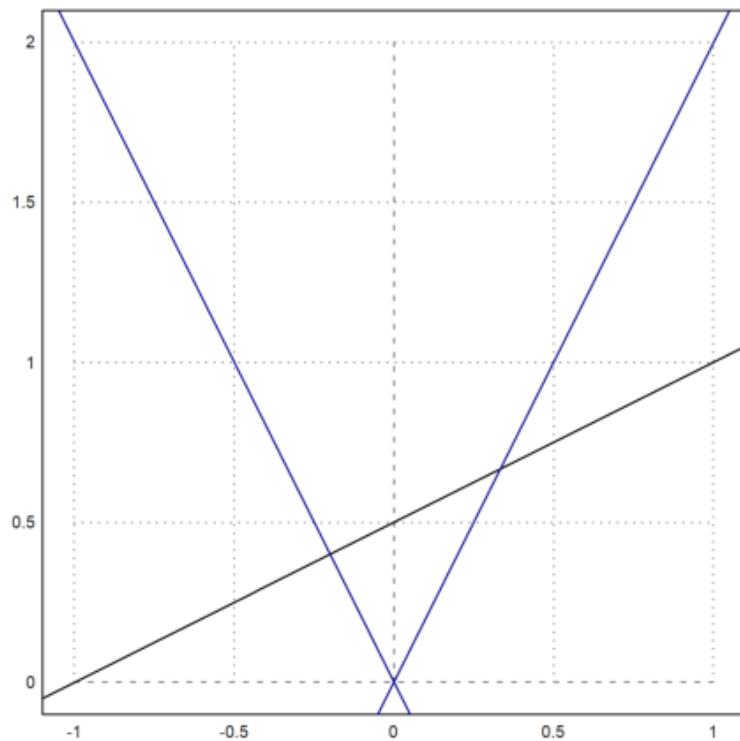
Lalu baris ketiga.

```
>g &= lineThrough([-1,0],[1,1])
```

```
[- 1, 2, 1]
```

Kita 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]
```

Hitung jarak ke g1.

```
>d1 &= distance(P,projectToLine(P,g1)); $d1
```

$$\sqrt{\left(\frac{a^2 u}{a^2 + 1} - u\right)^2 + \frac{a^2 u^2}{(a^2 + 1)^2}}$$

Hitung jarak ke g.

```
>d &= distance(P,projectToLine(P,g)); $d
```

$$\sqrt{\left(\frac{u+2}{5} - u\right)^2 + \frac{(2u-1)^2}{25}}$$

Dan temukan pusat kedua lingkaran, di mana jaraknya sama.

```
>sol &= solve(d1^2=d^2,u); $sol
```

$$\left[u = \frac{-\sqrt{5}\sqrt{a^2+1} + 2a^2 + 2}{4a^2 - 1}, u = \frac{\sqrt{5}\sqrt{a^2+1} + 2a^2 + 2}{4a^2 - 1} \right]$$

Ada dua solusi.

Kami mengevaluasi solusi simbolis, dan menemukan kedua pusat, dan kedua jarak.

```
>u := sol()
```

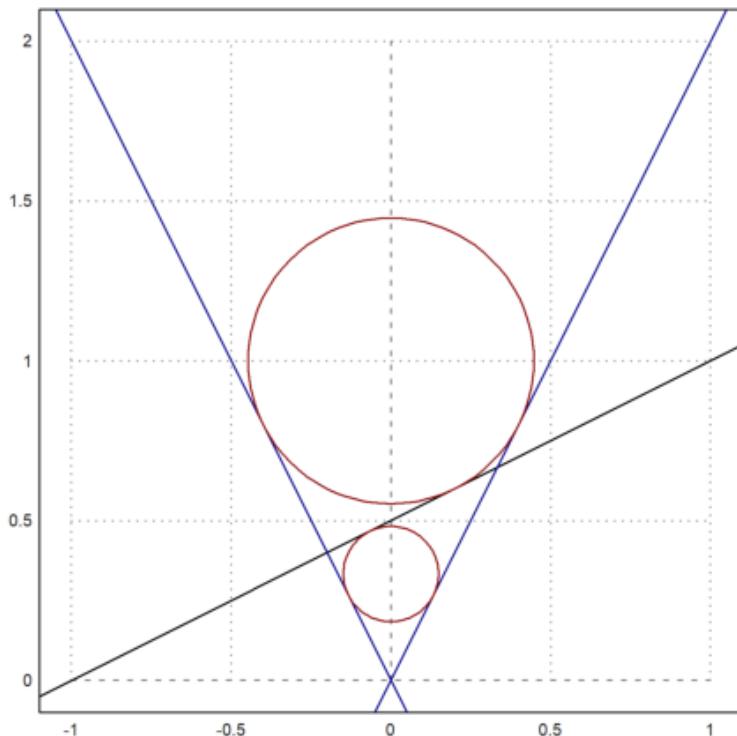
```
[0.333333, 1]
```

```
>dd := d()
```

```
[0.149071, 0.447214]
```

Plot lingkaran ke dalam gambar.

```
>color(red);
>plotCircle(circleWithCenter([0,u[1]],dd[1]), "");
>plotCircle(circleWithCenter([0,u[2]],dd[2]), "");
>insimg;
```



Plot dengan Povray

Selanjutnya kami merencanakan semuanya dengan Povray. Perhatikan bahwa Anda mengubah perintah apa pun dalam urutan perintah Povray berikut, dan menjalankan 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

Kita mengatur adegan dengan tepat.

```
>povstart(zoom=11,center=[0,0,0.5],height=10°,angle=140°);
```

Selanjutnya kita menulis dua bidang 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 pesawat 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 kami menghasilkan dua titik pada lingkaran, di mana bola menyentuh kerucut.

```
>function turnz(v) := return [-v[2],v[1],v[3]]  
>P1=projectToLine([0,u[1]],g()); P1=turnz([P1[1],0,P1[2]]);  
>writeln(povpoint(P1,povlook(yellow)));  
>P2=projectToLine([0,u[2]],g()); P2=turnz([P2[1],0,P2[2]]);  
>writeln(povpoint(P2,povlook(yellow)));
```

Kemudian kami menghasilkan dua titik di mana bola 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 dengan segmen garis.

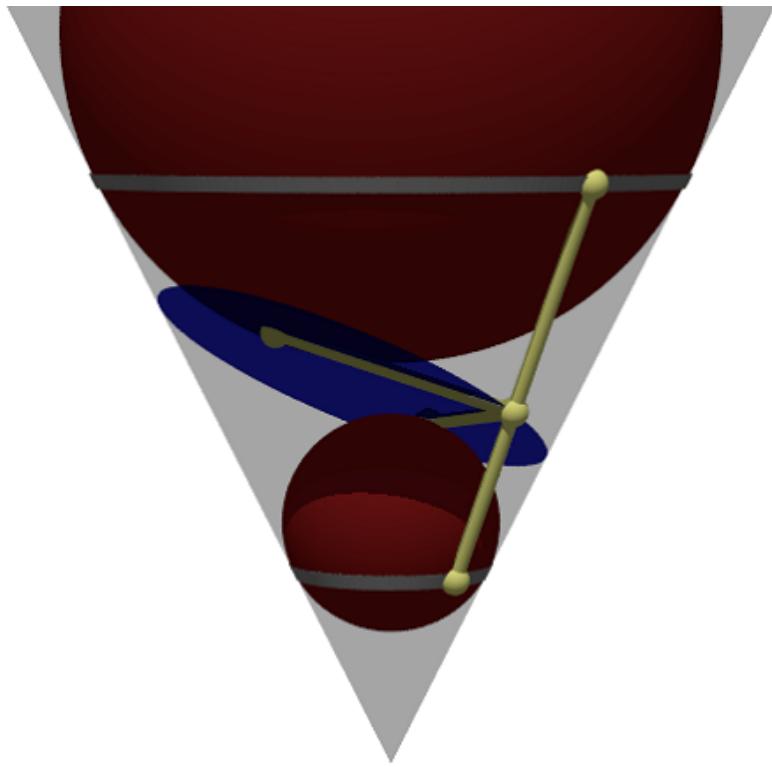
```
>writeln(povsegment(P1,P2,povlook(yellow)));  
>writeln(povsegment(P5,P3,povlook(yellow)));  
>writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang kami membuat pita abu-abu, di mana bola menyentuh kerucut.

```
>pcw=povcone([0,0,0],0,[0,0,a],1.01);  
>pc1=povcylinder([0,0,P1[3]-defaultpointsiz/2],[0,0,P1[3]+defaultpointsiz/2],1);  
>writeln(povintersection([pcw,pc1],povlook(gray)));  
>pc2=povcylinder([0,0,P2[3]-defaultpointsiz/2],[0,0,P2[3]+defaultpointsiz/2],1);  
>writeln(povintersection([pcw,pc2],povlook(gray)));
```

Mulai program Povray.

```
>povend();
```



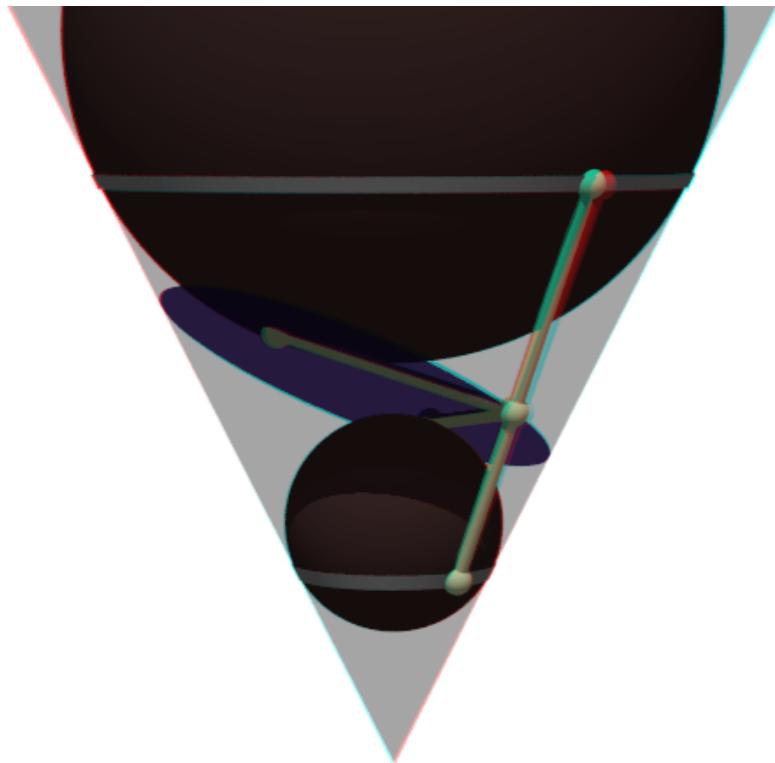
Untuk mendapatkan Anaglyph ini, kita perlu memasukkan semuanya ke dalam fungsi scene. Fungsi ini akan digunakan dua kali nanti.

```
>function scene () ...
global a,u,dd,g,g1,defaultpointsize;
writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));
writeln(povsphere([0,0,u[2]],dd[2],povlook(red)));
writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)));
gp=g();
pc=povcone([0,0,0],0,[0,0,a],1,"");
vp=[gp[1],0,gp[2]]; dp=gp[3];
writeln(povplane(vp,dp,povlook(blue,0.5),pc));
P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]);
writeln(povpoint(P1,povlook(yellow)));
P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]);
writeln(povpoint(P2,povlook(yellow)));
P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]];
writeln(povpoint(P3,povlook(yellow)));
P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]];
writeln(povpoint(P4,povlook(yellow)));
t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-t2)*(P2-P1);
writeln(povpoint(P5,povlook(yellow)));
writeln(povsegment(P1,P2,povlook(yellow)));
writeln(povsegment(P5,P3,povlook(yellow)));
writeln(povsegment(P5,P4,povlook(yellow)));
pcw=povcone([0,0,0],0,[0,0,a],1.01);
pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2],1);
writeln(povintersection([pcw,pc1],povlook(gray)));
pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsize/2],1);
```

```
writeln(povintersection([pcw,pc2],povlook(gray)));
endfunction
```

Anda membutuhkan kacamata merah / cyan untuk mengapresiasi efek berikut.

```
>povanaglyph("scene", zoom=11, center=[0, 0, 0.5], height=10°, angle=140°);
```



Contoh 8: Geometri Bumi

Di notebook ini, kami ingin melakukan beberapa komputasi bola. Fungsi-fungsi tersebut terdapat dalam file "spherical.e" di folder contoh. Kita perlu memuat file itu dulu.

```
>load "spherical.e";
```

Untuk memasukkan posisi geografis, kami menggunakan vektor dengan dua koordinat dalam radian (utara dan timur, nilai negatif untuk selatan dan barat). Berikut koordinat Kampus FMIPA UNY.

```
>FMIPA=[rad(-7,-46.467),rad(110,23.05)]
```

```
[-0.13569, 1.92657]
```

Anda dapat mencetak posisi ini dengan sposprint (cetak posisi bola).

```
>sposprint(FMIPA) // posisi garis lintang dan garis bujur FMIPA UNY
```

```
S 7°46.467' E 110°23.050'
```

Mari kita tambahkan dua kota lagi, 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 bola ke bola lainnya pada bola ideal. Vektor ini adalah [heading, distance] dalam radian. Untuk menghitung jarak di bumi, kita mengalikan dengan jari-jari bumi pada garis lintang 7° .

```
>br=svector(FMIPA,Solo); degprint(br[1]), br[2]*rearth(7°)->km // perkiraan jarak FMIPA-Solo
```

```
65°20'26.60''  
53.8945384608
```

Ini adalah perkiraan yang bagus. Rutinitas berikut menggunakan perkiraan yang lebih baik. Pada jarak yang begitu dekat hasilnya hampir sama.

```
>esdist(FMIPA,Semarang)->" km", // perkiraan jarak FMIPA-Semarang
```

```
88.0114026318 km
```

Ada fungsi untuk heading, dengan mempertimbangkan bentuk bumi yang elips. 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);
```

```
180°0'10.77''
```

Ini dapat digunakan untuk menghitung luas segitiga. Catatan: Untuk segitiga kecil, ini tidak akurat karena kesalahan pengurangan dalam asum- π .

```
>(asum- $\pi$ )*rearth(48°)^2->" km^2", // perkiraan luas segitiga FMIPA-Solo-Semarang
```

```
2116.02948749 km^2
```

Ada fungsi untuk ini, yang menggunakan garis lintang rata-rata segitiga untuk menghitung jari-jari bumi, dan menangani kesalahan pembulatan untuk segitiga yang sangat kecil.

```
>esarea(Solo,FMIPA,Semarang)->" km^2", //perkiraan yang sama dengan fungsi esarea()
```

```
2123.64310526 km^2
```

Kami juga dapat menambahkan vektor ke posisi. Vektor berisi heading dan jarak, keduanya dalam radian. Untuk mendapatkan vektor, kami menggunakan svector. Untuk menambahkan vektor ke posisi, kami 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 429,66 km. Kami mendapatkan perkiraan yang bagus.

```
>esdist(Tugu,Monas)->" km", // perkiraan jarak Tugu Jogja - Monas Jakarta
```

```
431.565659488 km
```

Judulnya sama dengan yang dihitung di Google Earth.

```
>degsprint(esdir(Tugu,Monas))
```

```
294°17'2.85''
```

Namun, kita tidak lagi mendapatkan posisi target yang tepat, jika kita menambahkan heading dan jarak ke posisi semula. Hal ini terjadi, karena kita tidak menghitung fungsi invers secara tepat, tetapi mengambil perkiraan jari-jari bumi di sepanjang jalan.

```
>sposprint(esadd(Tugu,esdir(Tugu,Monas),esdist(Tugu,Monas)))
```

```
S 6°10.500' E 106°48.717'
```

Namun, kesalahannya tidak besar.

```
>sposprint(Monas),
```

```
S 6°10.500' E 106°48.717'
```

Tentunya kita tidak bisa berlayar dengan tujuan yang sama dari satu tujuan ke tujuan lainnya, jika kita ingin mengambil jalur terpendek. Bayangkan, Anda terbang NE mulai dari titik mana pun di bumi. Kemudian Anda akan berputar ke kutub utara. Lingkaran besar tidak mengikuti arah yang konstan! Perhitungan berikut menunjukkan bahwa kami jauh dari tujuan yang benar, jika kami menggunakan tajuk yang sama selama perjalanan kami.

```
>dist=esdist(Tugu,Monas); hd=esdir(Tugu,Monas);
```

Sekarang kita tambahkan 10 kali sepersepuluh jaraknya, menggunakan heading ke Monas, kita sampai di Tugu.

```
>p=Tugu; loop 1 to 10; p=esadd(p,hd,dist/10); end;
```

Hasilnya masih jauh.

```
>sposprint(p), skmpprint(esdist(p,Monas))
```

```
S 6°11.250' E 106°48.372'  
1.529km
```

Sebagai contoh lain, mari kita ambil dua titik di bumi pada ketinggian yang sama.

```
>P1=[30°,10°]; P2=[30°,50°];
```

Jalur terpendek dari P1 ke P2 bukanlah lingkaran dengan garis lintang 30° , tetapi jalur yang lebih pendek mulai 10° lebih jauh ke utara di P1.

```
>sdegprint(esdir(P1,P2))
```

```
79.69°
```

Tapi, jika kita mengikuti pembacaan kompas ini, kita akan berputar ke kutub utara! Jadi kita harus menyesuaikan arah tujuan kita di sepanjang jalan. Untuk tujuan kasar, kami 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 tajuk yang sama terlalu lama.

```
>skmprint(esdist(p,P2))
```

0.203km

Kami mendapatkan perkiraan yang baik, jika kami menyesuaikan heading setelah setiap 1/100 dari total jarak dari Tugu ke Monas.

```
>p=Tugu; dist=esdist(Tugu,Monas); ...
> loop 1 to 100; p=esadd(p,esdir(p,Monas),dist/100); end;
>skmprint(esdist(p,Monas))
```

0.000km

Untuk keperluan navigasi, kita bisa mendapatkan urutan posisi GPS di sepanjang lingkaran besar menuju Monas dengan fungsi navigasi.

```
>load spherical; v=navigate(Tugu,Monas,10); ...
> loop 1 to rows(v); sposprint(v[#]), end;
```

```
S 7°46.998' E 110°21.966'
S 7°37.422' E 110°0.573'
S 7°27.829' E 109°39.196'
S 7°18.219' E 109°17.834'
S 7°8.592' E 108°56.488'
S 6°58.948' E 108°35.157'
S 6°49.289' E 108°13.841'
S 6°39.614' E 107°52.539'
S 6°29.924' E 107°31.251'
S 6°20.219' E 107°9.977'
S 6°10.500' E 106°48.717'
```

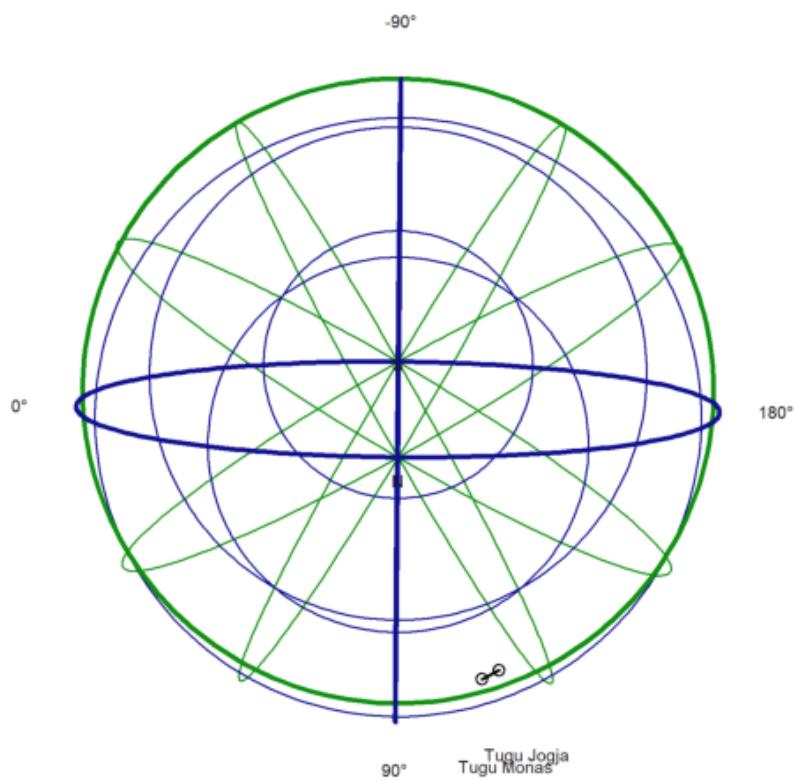
Kami menulis sebuah fungsi, yang menggambarkan bumi, dua posisi, dan posisi di antaranya.

```
>function testplot ...
```

```
useglobal;
plotearth;
plotpos(Tugu,"Tugu Jogja"); plotpos(Monas,"Tugu Monas");
plotposline(v);
endfunction
```

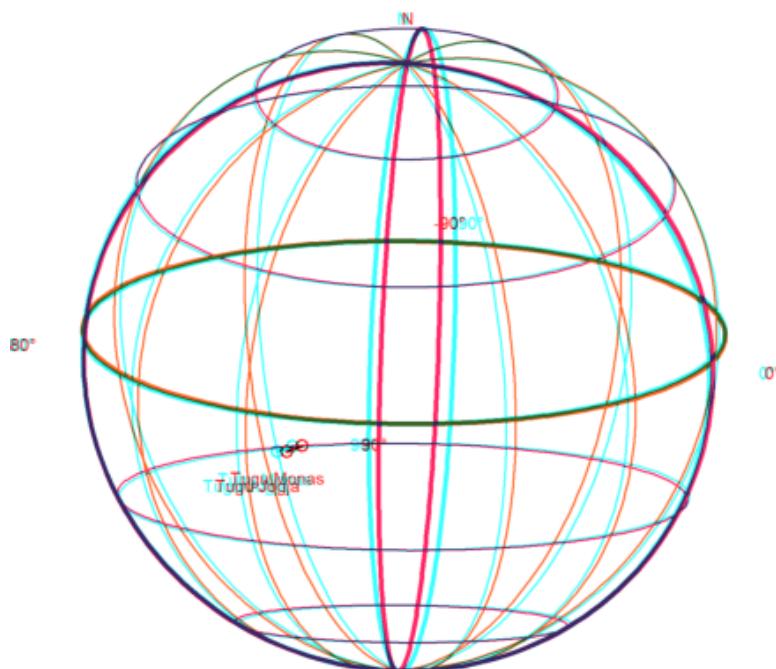
Sekarang plot semuanya.

```
>plot3d("testplot",angle=25, height=6,>own,>user,zoom=4):
```



Atau gunakan `plot3d` untuk mendapatkan tampilan anaglyphnya. Ini terlihat sangat bagus dengan kacamata merah / cyan.

```
>plot3d("testplot",angle=25,height=6,distance=5,own=1,anaglyph=1,zoom=4) :
```



Latihan

1. Gambarlah segi-n beraturan jika diketahui titik pusat O, n, dan jarak titik pusat ke titik-titik sudut segi-n tersebut (jari-jari lingkaran luar segi-n), r.

Petunjuk:

- Besar sudut pusat yang menghadap masing-masing sisi segi-n adalah $(360/n)$.
- Titik-titik sudut segi-n merupakan perpotongan lingkaran luar segi-n dan garis-garis yang melalui pusat dan saling membentuk sudut sebesar kelipatan $(360/n)$.
- Untuk n ganjil, pilih salah satu titik sudut adalah di atas.
- Untuk n genap, pilih 2 titik di kanan dan kiri lurus dengan titik pusat.
- Anda dapat menggambar segi-3, 4, 5, 6, 7, dst beraturan.

2. Gambarlah suatu parabola yang melalui 3 titik yang diketahui.

Petunjuk:

- Misalkan persamaan parabolanya $y = ax^2 + bx + c$.
- Substitusikan koordinat titik-titik yang diketahui ke persamaan tersebut.
- Selesaikan SPL yang terbentuk untuk mendapatkan nilai-nilai a, b, c.

3. Gambarlah suatu segi-4 yang diketahui keempat titik sudutnya, misalnya A, B, C, D.

– Tentukan apakah segi-4 tersebut merupakan segi-4 garis singgung

(sisinya-sisinya merupakan garis singgung lingkaran yang sama yakni lingkaran dalam segi-4 tersebut).

– Suatu segi-4 merupakan segi-4 garis singgung apabila keempat

garis bagi sudutnya bertemu di satu titik.

– Jika segi-4 tersebut merupakan segi-4 garis singgung, gambar

lingkaran dalamnya.

– Tunjukkan bahwa syarat suatu segi-4 merupakan segi-4 garis

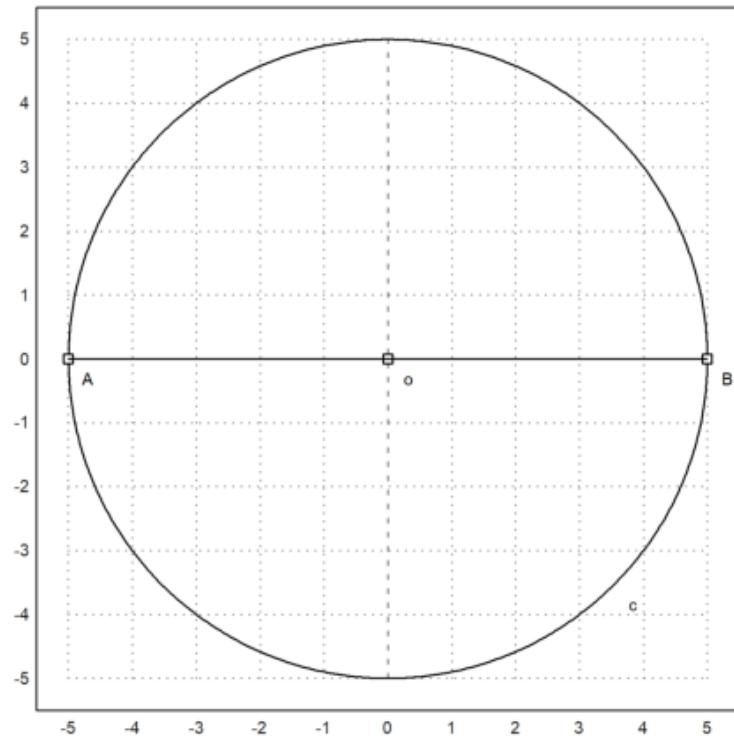
singgung apabila hasil kali panjang sisi-sisi yang berhadapan sama.

4. Gambarlah suatu ellips jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang jumlah jarak ke P dan ke Q selalu sama (konstan).

5. Gambarlah suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).

Jawab :

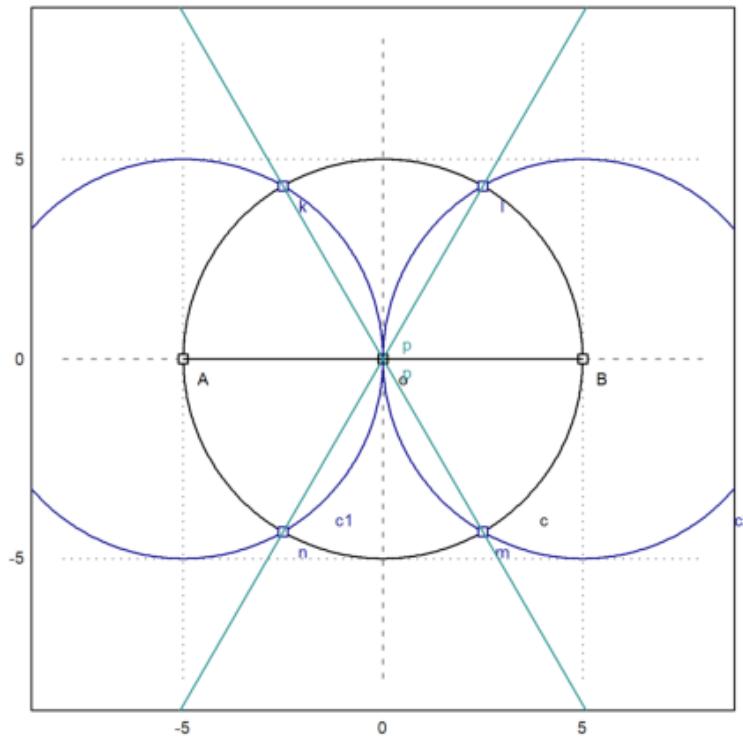
```
>o &:= [0,0]; c=circleWithCenter(o,5);
>color(1); setPlotRange(5); plotPoint(o); plotCircle(c);
>A=[-5,0]; plotPoint(A,"A");
>B=[5,0]; plotPoint(B,"B");
>plotSegment(A,B,"");
```



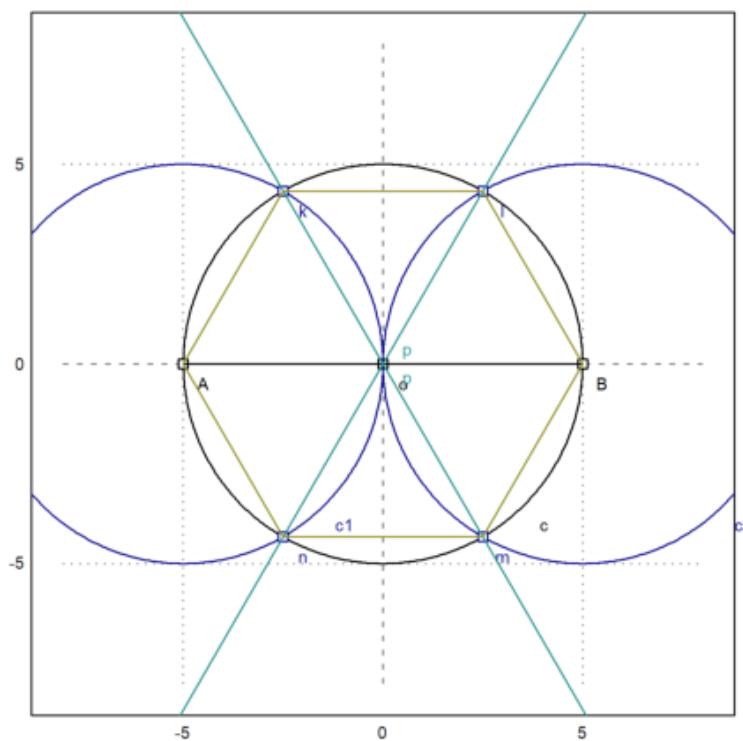
```

>c1=circleWithCenter(A,distance(A,o));
>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(r,s);
>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,"");
```



BAB 7

KB PEKAN 10; MENGGUNAKAN EMT UNTUK STATISTIKA

article
eumat

EMT untuk Statistika

Nama : Shalih Abdillah
NIM : 22305144009
Kelas : Matematika B 2022

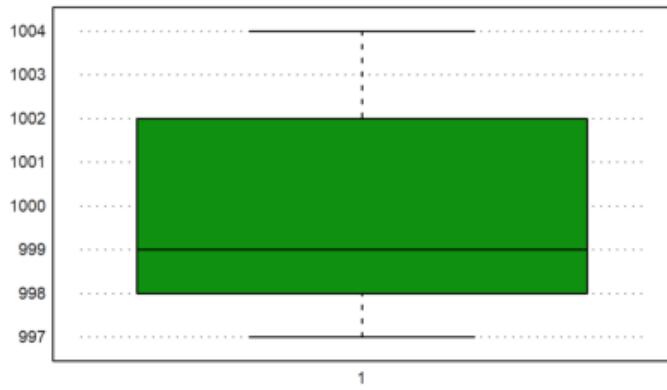
Dalam buku catatan ini, kami mendemonstrasikan plot statistik utama, tes, dan distribusi dalam Euler. Mari kita mulai dengan beberapa statistik deskriptif. Ini bukanlah sebuah pengantar statistik. Jadi, Anda mungkin memerlukan latar belakang untuk memahami detailnya.
Asumsikan pengukuran berikut. Kita ingin menghitung nilai rata-rata dan deviasi standar yang diukur.

```
>M=[1000,1004,998,997,1002,1001,998,1004,998,997]; ...  
>median(M), mean(M), dev(M),
```

999
999.9
2.72641400622

Kita dapat memplot plot kotak dan kumis untuk data tersebut. Dalam kasus kami, tidak ada pencilan.

```
>aspect(1.75); boxplot(M);
```



Kami menghitung probabilitas bahwa suatu nilai lebih besar dari 1005, dengan mengasumsikan nilai yang diukur dari distribusi normal.

Semua fungsi untuk distribusi dalam Euler diakhiri dengan ...dis dan menghitung distribusi probabilitas kumulatif (CPF).

$$\text{normaldis}(x, m, d) = \int_{-\infty}^x \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t-m}{d})^2} dt.$$

Kami mencetak hasilnya dalam % dengan akurasi 2 digit menggunakan fungsi cetak.

```
>print ((1-normaldis(1005,mean(M),dev(M)))*100,2,unit=" %")
```

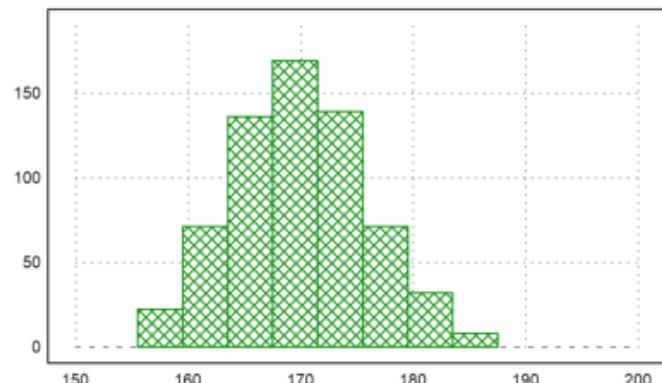
3.07 %

Untuk contoh berikutnya, kami mengasumsikan jumlah pria berikut ini dalam rentang ukuran tertentu.

```
>r=155.5:4:187.5; v=[22,71,136,169,139,71,32,8];
```

Berikut ini adalah plot distribusinya.

```
>plot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style="/"):
```



Kita dapat memasukkan data mentah tersebut ke dalam tabel.

Tabel adalah sebuah metode untuk menyimpan data statistik. Tabel kita harus berisi tiga kolom: Awal rentang, akhir rentang, jumlah orang dalam rentang.

Tabel dapat dicetak dengan header. Kami menggunakan vektor string untuk mengatur header.

```
>T:=r[1:8]' | r[2:9]' | v'; writetable(T, labc=["BB", "BA", "Frek"])
```

BB	BA	Frek
155.5	159.5	22
159.5	163.5	71
163.5	167.5	136
167.5	171.5	169
171.5	175.5	139
175.5	179.5	71
179.5	183.5	32
183.5	187.5	8

Jika kita membutuhkan nilai rata-rata dan statistik lain dari ukuran, kita perlu menghitung titik tengah rentang. Kita dapat menggunakan dua kolom pertama dari tabel kita untuk hal ini.

Sumbol "|" digunakan untuk memisahkan kolom, fungsi "writetable" digunakan untuk menulis tabel, dengan opsi "labc" untuk menentukan judul kolom.

```
>(T[,1]+T[,2])/2 // the midpoint of each interval
```

```
157.5  
161.5  
165.5  
169.5  
173.5  
177.5  
181.5  
185.5
```

Tetapi akan lebih mudah, untuk melipat rentang dengan vektor [1/2,1/2].

```
>M=fold(r, [0.5,0.5])
```

```
[157.5, 161.5, 165.5, 169.5, 173.5, 177.5, 181.5, 185.5]
```

Sekarang kita dapat menghitung rata-rata dan deviasi sampel dengan frekuensi yang diberikan.

```
>{m, d}=meandev(M, v); m, d,
```

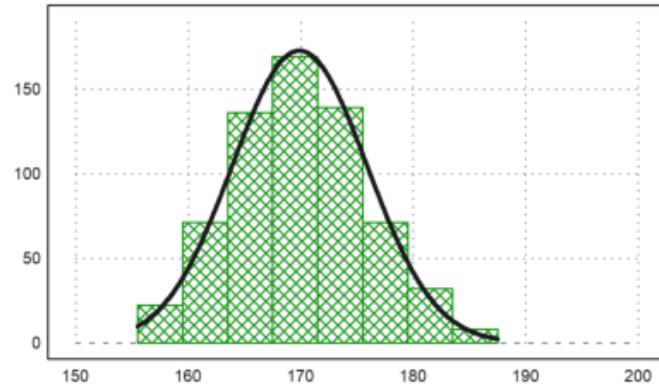
```
169.901234568  
5.98912964449
```

Mari kita tambahkan distribusi normal dari nilai-nilai tersebut ke dalam diagram batang di atas. Rumus untuk distribusi normal dengan rata-rata m dan deviasi standar d adalah:

$$y = \frac{1}{d\sqrt{2\pi}} e^{\frac{-(x-m)^2}{2d^2}}.$$

Karena nilainya antara 0 dan 1, untuk memplotnya pada diagram batang, nilai tersebut harus dikalikan dengan 4 kali jumlah data.

```
>plot2d("qnormal(x,m,d)*sum(v)*4", ...
> xmin=min(r),xmax=max(r),thickness=3,add=1):
```



Tables

Dalam direktori buku catatan ini, Anda akan menemukan file dengan tabel. Data tersebut merupakan hasil survei. Berikut adalah empat baris pertama dari file tersebut. Data berasal dari sebuah buku online berbahasa Jerman "Einführung in die Statistik mit R" oleh A. Handl.

```
>printfile("table.dat", 4);
```

Person	Sex	Age	Titanic	Evaluation	Tip	Problem
1	m	30	n	.	1.80	n
2	f	23	y	g	1.80	n
3	f	26	y	g	1.80	y

Tabel berisi 7 kolom angka atau token (string). Kita ingin membaca tabel tersebut dari file. Pertama, kita menggunakan terjemahan kita sendiri untuk token-token tersebut.

Untuk itu, kita mendefinisikan set token. Fungsi `strtokens()` mendapatkan vektor string token dari string yang diberikan.

```
>mf:=[ "m", "f" ]; yn:=[ "y", "n" ]; ev:=strtokens("g vg m b vb");
```

Sekarang kita membaca tabel dengan terjemahan ini.

Argumen `tok2`, `tok4`, dan lain-lain adalah terjemahan dari kolom-kolom tabel. Argumen-argumen ini tidak ada dalam daftar parameter `readtable()`, jadi Anda harus menyediakannya dengan :=.

```
>{MT,hd}=readtable("table.dat",tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
>load over statistics;
```

Untuk mencetak, kita perlu menentukan set token yang sama. Kami mencetak empat baris pertama saja.

```
>writetable(MT[1:10],labc=hd,wc=5,tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
```

Person	Sex	Age	Titanic	Evaluation	Tip	Problem
1	m	30	n	.	1.8	n
2	f	23	y	g	1.8	n
3	f	26	y	g	1.8	y
4	m	33	n	.	2.8	n
5	m	37	n	.	1.8	n
6	m	28	y	g	2.8	y
7	f	31	y	vg	2.8	n
8	m	23	n	.	0.8	n
9	f	24	y	vg	1.8	y
10	m	26	n	.	1.8	n

Tanda titik "." mewakili nilai yang tidak tersedia.

Jika kita tidak ingin menentukan token untuk terjemahan sebelumnya, kita hanya perlu menentukan kolom mana yang berisi token dan bukan angka.

```
>ctok=[2,4,5,7]; {MT,hd,tok}=readtable("table.dat",ctok=ctok);
```

Fungsi readtable() sekarang mengembalikan satu set token.

```
>tok
```

```
m  
n  
f  
y  
g  
vg
```

Tabel berisi entri dari file dengan token yang diterjemahkan ke angka.

String khusus NA="." ditafsirkan sebagai "Tidak Tersedia", dan mendapatkan NAN (bukan angka) dalam tabel. Terjemahan ini dapat diubah dengan parameter NA, dan NAvl.

```
>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, melewatkannya 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 menggunakan untuk menentukan nilai rata-rata kolom atau nilai statistik lainnya.

```
>mean(tablecol(MT,6))
```

2.175

Fungsi `getstatistics()` mengembalikan elemen-elemen dalam sebuah vektor, dan jumlahnya. Kita menerapkannya pada nilai "m" dan "f" pada kolom kedua tabel kita.

```
>{xu,count}=getstatistics(tablecol(MT,2)); xu, count,
```

[1, 3]
[12, 13]

Kita bisa mencetak hasilnya dalam tabel baru.

```
>writetable(count',labr=tok[xu])
```

m	12
f	13

Fungsi `selectable()` 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
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

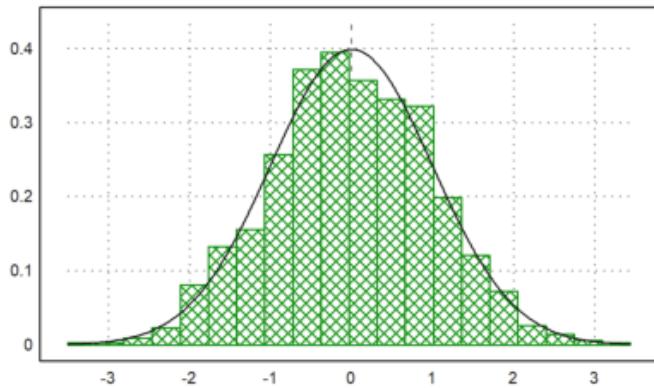
Dan hapus file tersebut.

```
>fileremove(filename);
```

Distribusi

Dengan `plot2d`, ada metode yang sangat mudah untuk memplot distribusi data eksperimen.

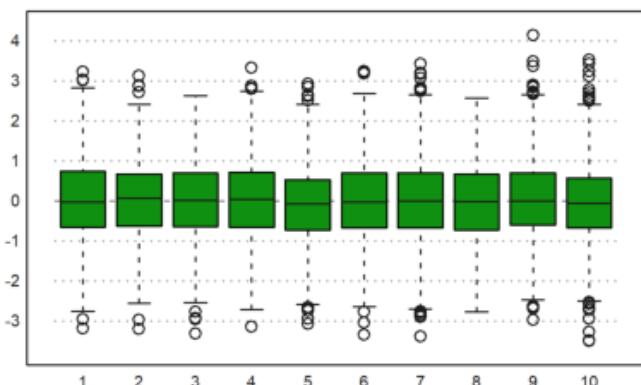
```
>p=normal(1,1000); //1000 random normal-distributed sample p
>plot2d(p,distribution=20,style="\\""); // plot the random sample p
>plot2d("qnormal(x,0,1)",add=1); // add the standard normal distribution plot
```



Perhatikan perbedaan antara plot batang (sampel) dan kurva normal (distribusi sesungguhnya). Masukkan kembali ketiga perintah tersebut untuk melihat hasil pengambilan sampel yang lain.

Berikut ini adalah perbandingan 10 simulasi dari 1000 nilai terdistribusi normal dengan menggunakan apa yang disebut plot kotak. Plot ini menunjukkan median, kuartil 25% dan 75%, nilai minimal dan maksimal, serta pencilan.

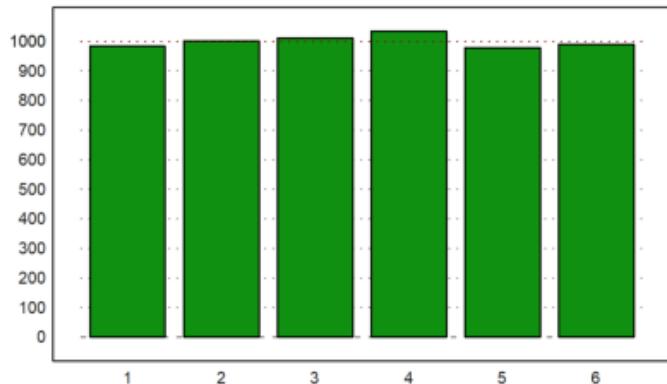
```
>p=normal(10,1000); boxplot(p):
```



Untuk menghasilkan bilangan bulat acak, Euler memiliki intrandom. Mari kita simulasikan pelemparan dadu dan memplot distribusinya.

Kita menggunakan fungsi getmultiplicities(v,x), yang menghitung seberapa sering elemen-elemen dari v muncul di dalam x. Kemudian kita memplot hasilnya menggunakan columnsplot().

```
>k=intrandom(1,6000,6); ...
>columnsplot(getmultiplicities(1:6,k)); ...
>ygrid(1000,color=red):
```



Meskipun intrandom(n,m,k) menghasilkan bilangan bulat yang terdistribusi secara seragam dari 1 sampai k, adalah mungkin untuk menggunakan distribusi bilangan bulat yang lain dengan randpint().

Pada contoh berikut, probabilitas untuk 1,2,3 adalah 0.4, 0.1, 0.5 secara berurutan.

```
>randpint(1,1000,[0.4,0.1,0.5]); getmultiplicities(1:3,%)
```

[378, 102, 520]

Euler dapat menghasilkan nilai acak dari lebih banyak distribusi. Lihatlah ke dalam referensi.

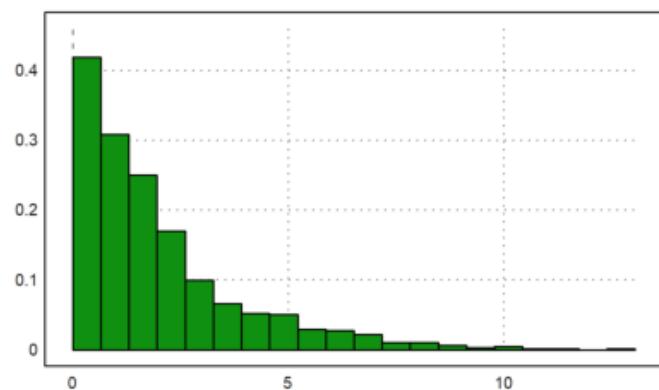
Misalnya, kita mencoba distribusi eksponensial. Sebuah variabel acak kontinu X dikatakan memiliki distribusi eksponensial, jika PDF-nya diberikan oleh

$$f_X(x) = \lambda e^{-\lambda x}, \quad x > 0, \quad \lambda > 0,$$

dengan parameter

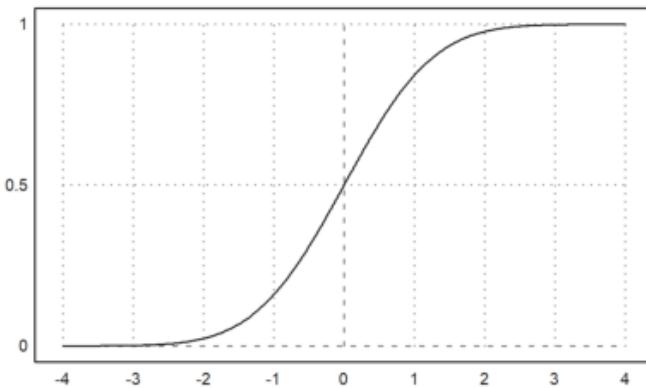
$$\lambda = \frac{1}{\mu}, \quad \mu \text{ is the mean, and denoted by } X \sim \text{Exponential}(\lambda).$$

```
>plot2d(randexponential(1,1000,2),>distribution):
```



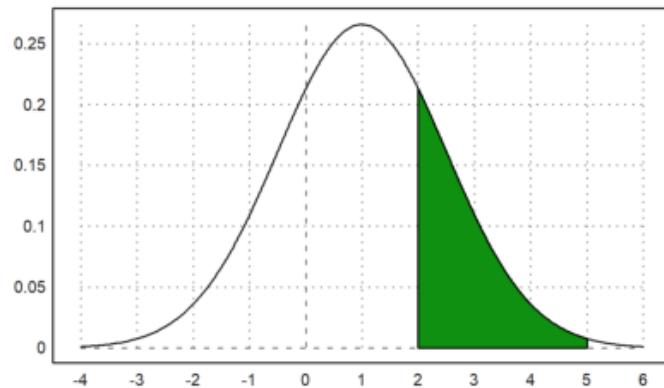
Untuk banyak distribusi, Euler dapat menghitung fungsi distribusi dan kebalikannya.

```
>plot2d("normaldis",-4,4):
```



Berikut ini adalah salah satu cara untuk memplot kuantil.

```
>plot2d("qnormal(x,1,1.5)",-4,6); ...  
>plot2d("qnormal(x,1,1.5)",a=2,b=5,>add,>filled):
```



$$\text{normaldis}(x,m,d) = \int_{-\infty}^x \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t-m}{d})^2} dt.$$

Probabilitas untuk berada di area hijau adalah sebagai berikut.

```
>normaldis(5,1,1.5)-normaldis(2,1,1.5)
```

0.248662156979

Hal ini dapat dihitung secara numerik dengan integral berikut ini.

$$\int_2^5 \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-1}{1.5})^2} dx.$$

```
>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.

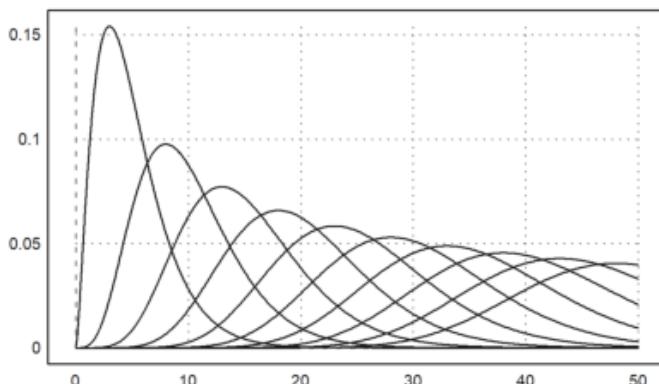
```
>invbindis(0.95,1000,0.5), invnormaldis(0.95,500,0.5*sqrt(1000))
```

525.516721219

526.007419394

Fungsi qdis() adalah densitas dari distribusi chi-square. Seperti biasa, Euler memetakan vektor ke fungsi ini. Dengan demikian kita mendapatkan plot semua distribusi chi-kuadrat dengan derajat 5 hingga 30 dengan mudah dengan cara berikut.

```
>plot2d("qchidis(x,(5:5:50)'),0,50):
```



Euler memiliki fungsi-fungsi yang akurat untuk mengevaluasi distribusi-distribusi. Mari kita periksa chidis() dengan sebuah integral.

Penamaannya diusahakan untuk konsisten. Sebagai contoh,

- distribusi chi-kuadrat adalah chidis(),
- fungsi kebalikannya adalah invchidis(),
- densitasnya adalah qchidis().

Pelengkap dari distribusi (ekor atas) adalah chicdis().

```
>chidis(1.5,2), integrate("qchidis(x,2)",0,1.5)
```

0.527633447259

0.527633447259

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]
```

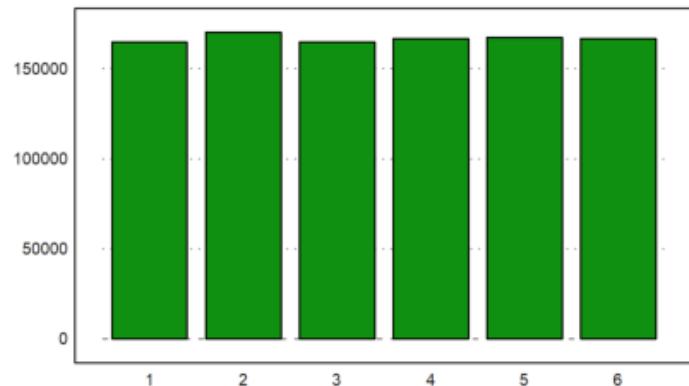
Artinya, dengan probabilitas $wd[i+1]-wd[i]$ kita menghasilkan nilai acak i.

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))):
```



Berikut ini adalah fungsi sederhana untuk memeriksa distribusi seragam dari nilai 1... K dalam v. Kami menerima hasilnya, jika untuk semua frekuensi

$$\left| f_i - \frac{1}{K} \right| < \frac{\delta}{\sqrt{n}}.$$

```
>function checkrandom (v, delta=1) ...
```

```
K=max(v); n=cols(v);
fr=getfrequencies(v,1:K);
return max(fr/n-1/K)<delta/sqrt(n);
endfunction
```

Memang fungsi ini menolak distribusi seragam.

```
>checkrandom(wrongdice(1,1000000))
```

0

Dan ini menerima generator acak bawaan.

```
>checkrandom(intrandom(1, 1000000, 6))
```

1

Kita dapat menghitung distribusi binomial. Pertama, ada `binomials()`, 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 `binomials()`.

```
>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

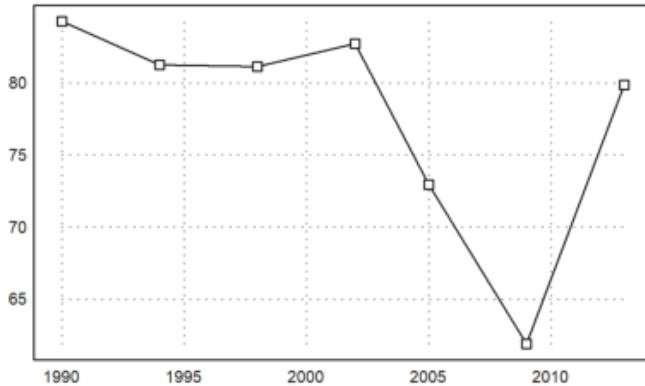
Perkalian matriks berikut ini mengekstrak jumlah persentase dua partai besar yang menunjukkan bahwa partai-partai kecil telah memperoleh suara di parlemen hingga tahun 2009.

```
>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 menggunakan garis dan titik secara bersamaan. Alternatif lainnya adalah memanggil plot2d dua kali dengan >add.

```
>statplot(YT,BT1,"b"):
```



Tentukan beberapa warna untuk masing-masing pihak.

```
>CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.8,0,0)];
```

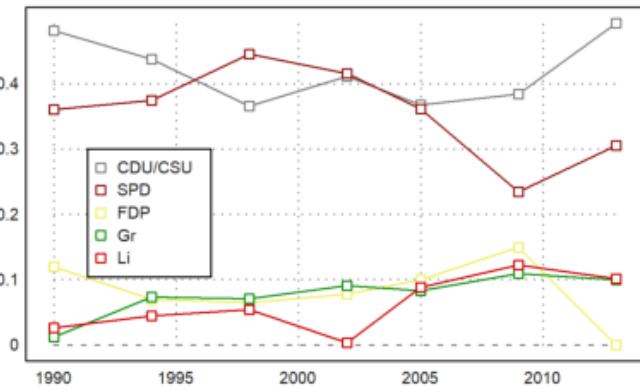
Sekarang kita dapat memplot hasil pemilu 2009 dan perubahannya ke dalam satu plot menggunakan figure. Kita dapat menambahkan vektor kolom pada setiap plot.

```
>figure(2,1); ...
>figure(1); columnsplot(BW[6,3:7],P,color=CP); ...
>figure(2); columnsplot(BW[6,3:7]-BW[5,3:7],P,color=CP); ...
>figure(0):
```



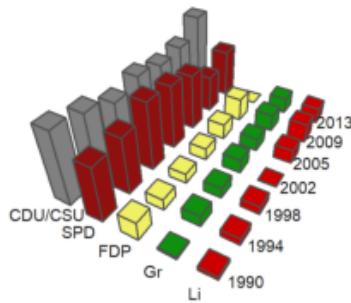
Plot data menggabungkan baris data statistik dalam satu plot.

```
>J:=BW[,1]'; DP:=BW[,3:7]'; ...
>dataplot(YT,BT',color=CP); ...
>labelbox(P,colors=CP,styles="[]",>points,w=0.2,x=0.3,y=0.4):
```



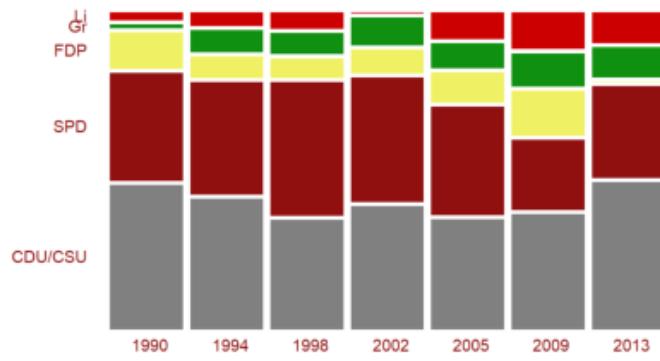
Plot kolom 3D menunjukkan deretan data statistik dalam bentuk kolom. Kami menyediakan label untuk baris dan kolom. angle adalah sudut pandang.

```
>columnspplot3d(BT, scols=P, srows=YT, ...
>  angle=30°, ccols=CP) :
```



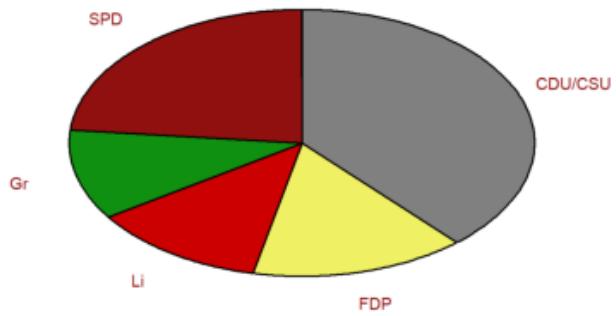
Representasi lainnya adalah plot mosaik. Perhatikan bahwa kolom-kolom pada plot mewakili kolom-kolom pada matriks di sini. Karena panjangnya label CDU/CSU, kita mengambil jendela yang lebih kecil dari biasanya.

```
>shrinkwindow(>smaller); ...
>mosaicplot(BT', srows=YT, scols=P, color=CP, style="#" ); ...
>shrinkwindow() :
```



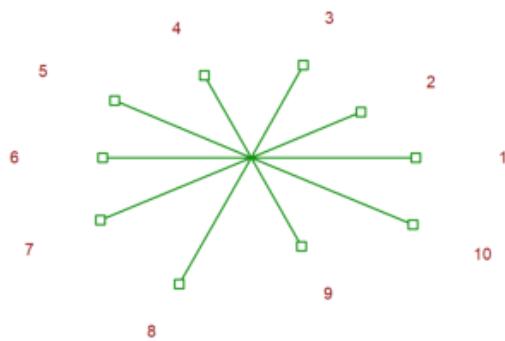
Kita juga bisa membuat diagram lingkaran. Karena warna hitam dan kuning membentuk sebuah koalisi, kita menyusun ulang elemen-elemennya.

```
>i=[1, 3, 5, 4, 2]; piechart(BW[6,3:7][i],color=CP[i],lab=P[i]):
```



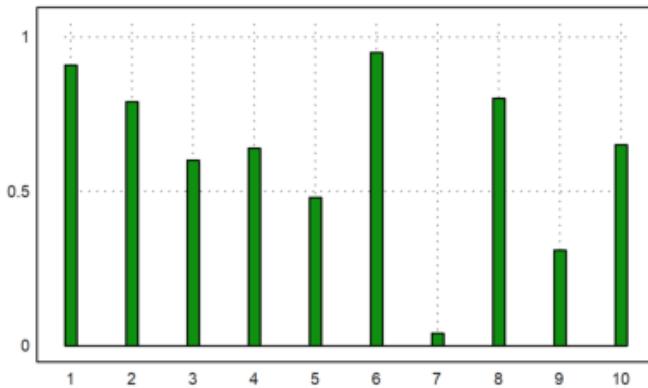
Berikut ini jenis plot yang lain.

```
>starplot(normal(1,10)+4,lab=1:10,>rays):
```



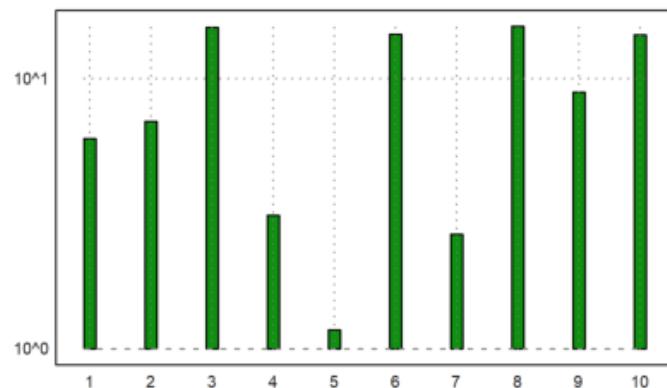
Beberapa plot di plot2d bagus untuk statika. Berikut ini adalah plot impuls dari data acak, yang terdistribusi secara seragam dalam [0,1].

```
>plot2d(makeimpulse(1:10,random(1,10)),>bar):
```



Tetapi untuk data yang terdistribusi secara eksponensial, kita mungkin memerlukan plot logaritmik.

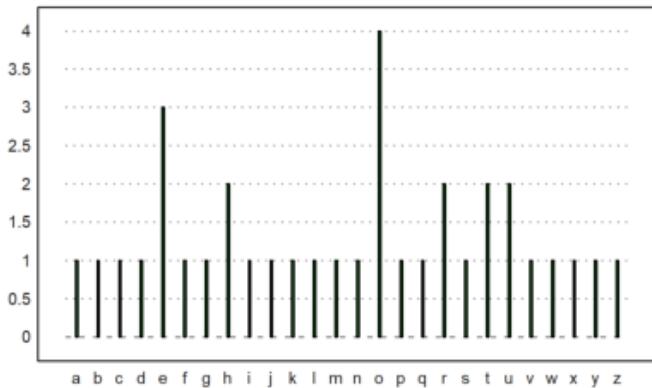
```
>logimpulseplot(1:10,-log(random(1,10))*10):
```



Fungsi columnsplot() lebih mudah digunakan, karena hanya membutuhkan sebuah vektor nilai. Selain itu, fungsi ini dapat mengatur labelnya menjadi apa pun yang kita inginkan, kita telah mendemonstrasikan hal ini dalam tutorial ini.

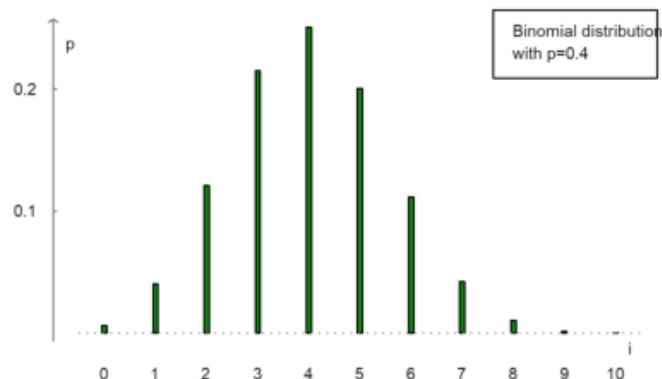
Berikut ini adalah aplikasi lain, di mana kita menghitung karakter dalam sebuah kalimat dan memplot statistik.

```
>v=strtochar("the quick brown fox jumps over the lazy dog"); ...
>w=ascii("a"):ascii("z"); x=getmultiplicities(w,v); ...
>cw=[]; for k=w; cw=cw|char(k); end; ...
>columnsplot(x,lab=cw,width=0.05):
```



Anda juga dapat menetapkan sumbu secara manual.

```
>n=10; p=0.4; i=0:n; x=bin(n,i)*p^i*(1-p)^(n-i); ...
>columnsplot(x,lab=i,width=0.05,<frame,<grid); ...
>yaxis(0,0:0.1:1,style="->",>left); xaxis(0,style="."); ...
>label("p",0,0.25), label("i",11,0); ...
>textbox(["Binomial distribution","with p=0.4"]):
```



Berikut ini adalah cara untuk memplot frekuensi angka dalam vektor. Kami membuat vektor angka acak bilangan bulat 1 hingga 6.

```
>v:=intrandom(1,10,10)
```

[8, 5, 8, 8, 6, 8, 8, 3, 5, 5]

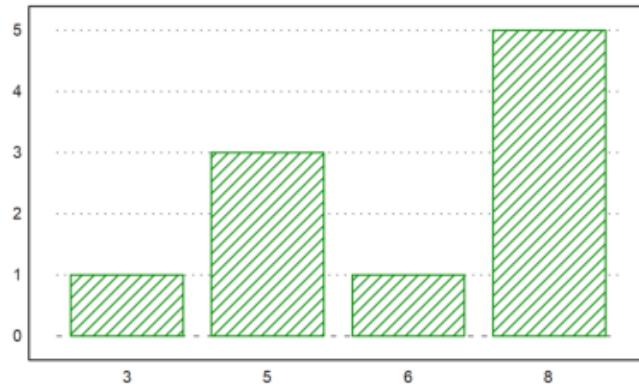
Kemudian ekstrak nomor unik dalam v.

```
>vu:=unique(v)
```

[3, 5, 6, 8]

Dan memplot frekuensi dalam plot kolom.

```
>columnsplot(getmultiplicities(vu,v),lab=vu,style="/"):
```



Kami ingin mendemonstrasikan fungsi untuk distribusi nilai empiris.

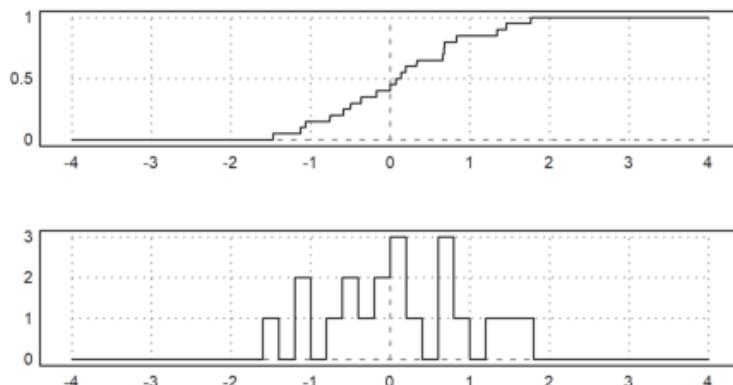
```
>x=normal(1,20);
```

Fungsi empdist(x,vs) membutuhkan larik nilai yang telah diurutkan. Jadi kita harus mengurutkan x sebelum dapat menggunakannya.

```
>xs=sort(x);
```

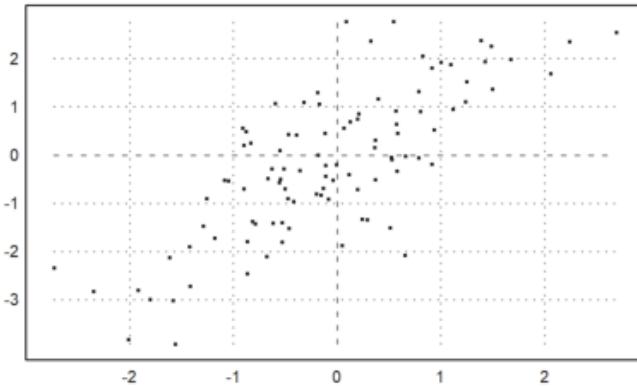
Kemudian kita memplot distribusi empiris dan beberapa batang kepadatan ke dalam satu plot. Alih-alih plot batang untuk distribusi, kali ini kami menggunakan plot gigi gergaji.

```
>figure(2,1); ...
>figure(1); plot2d("empdist",-4,4;xs); ...
>figure(2); plot2d(histo(x,v=-4:0.2:4,<bar)); ...
>figure(0):
```



Plot sebaran mudah dilakukan di Euler dengan plot titik biasa. Grafik berikut ini menunjukkan bahwa X dan X+Y berkorelasi positif secara jelas.

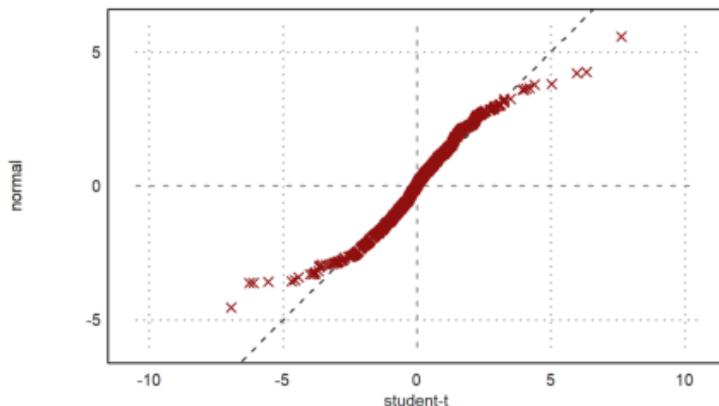
```
>x=normal(1,100); plot2d(x,x+rotright(x),>points,style="."):
```



Sering kali, kita ingin membandingkan dua sampel dari distribusi yang berbeda. Hal ini dapat dilakukan dengan plot kuantil-kuantil.

Untuk pengujian, kami mencoba distribusi student-t dan distribusi eksponensial.

```
>x=randt(1,1000,5); y=randnormal(1,1000,mean(x),dev(x)); ...
>plot2d("x",r=6,style="--",yl="normal",xl="student-t",>vertical); ...
>plot2d(sort(x),sort(y),>points,color=red,style="x",>add):
```



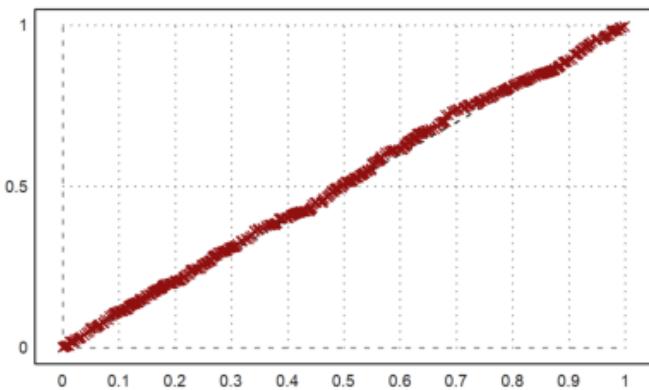
Plot ini dengan jelas menunjukkan bahwa nilai yang terdistribusi normal cenderung lebih kecil pada ujung yang ekstrim.

Jika kita memiliki dua distribusi dengan ukuran yang berbeda, kita dapat memperluas distribusi yang lebih kecil atau memperkecil distribusi yang lebih besar. Fungsi berikut ini bagus untuk keduanya. Fungsi ini mengambil nilai median dengan persentase antara 0 dan 1.

```
>function medianexpand (x,n) := median(x,p=linspace(0,1,n-1));
```

Mari kita bandingkan dua distribusi yang sama.

```
>x=random(1000); y=random(400); ...
>plot2d("x",0,1,style="--"); ...
>plot2d(sort(medianexpand(x,400)),sort(y),>points,color=red,style="x",>add) :
```



Regresi 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]

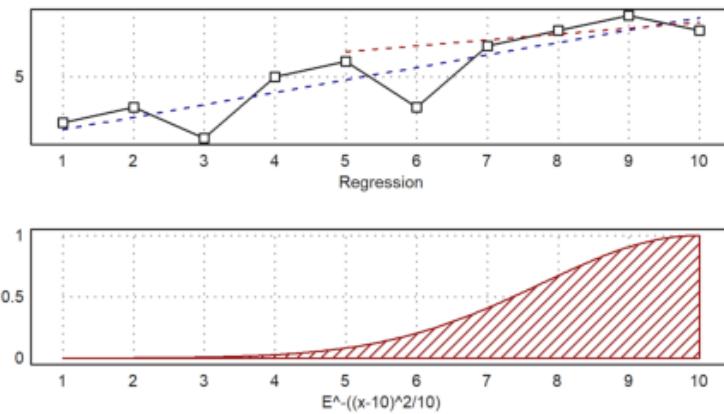
Sekarang, koefisien dengan bobot yang menekankan nilai terakhir.

```
>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.

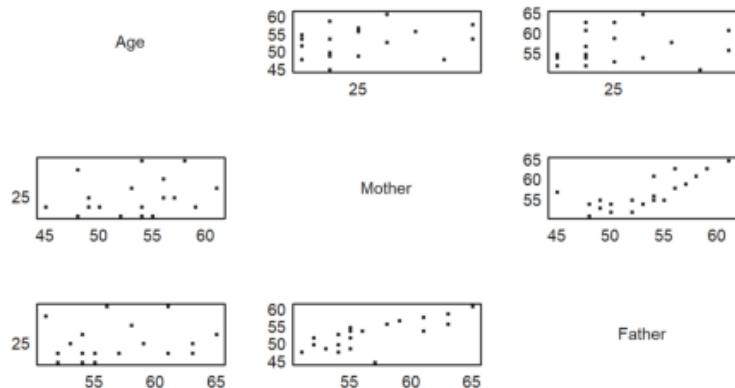
Tabel ini berisi "m" dan "f" pada kolom kedua. Kita menggunakan variabel tok2 untuk mengatur terjemahan yang tepat dan bukannya membiarkan readtable() mengumpulkan terjemahan.

```
>{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]
```

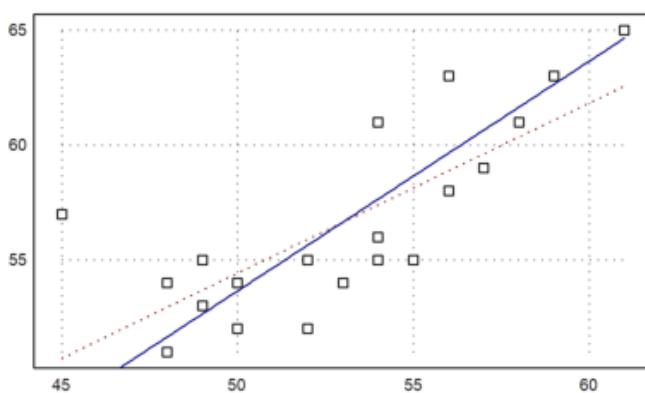
Ini jelas merupakan model yang salah. Garis regresinya adalah $s = 17 + 0,74t$, di mana t adalah usia ibu dan s adalah usia ayah. Perbedaan usia mungkin sedikit bergantung pada usia, tetapi tidak terlalu banyak. Sebaliknya, kami menduga fungsi seperti $s = a + t$. Kemudian a adalah rata-rata dari $s-t$. Ini adalah perbedaan usia rata-rata antara ayah dan ibu.

```
>da:=mean(cs[2]-cs[1])
```

```
3.65
```

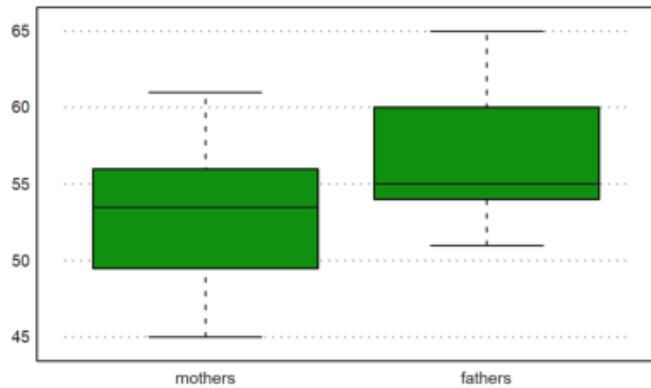
Mari kita plotkan ini ke dalam satu scatter plot.

```
>plot2d(cs[1], cs[2], >points); ...
>plot2d("evalpoly(x,ps)", color=red, style=".",>add); ...
>plot2d("x+da", color=blue,>add):
```



Berikut ini adalah plot kotak dari kedua usia tersebut. Ini hanya menunjukkan, bahwa usia keduanya berbeda.

```
>boxplot(cs, ["mothers", "fathers"]):
```



Sangat menarik bahwa perbedaan dalam median tidak sebesar perbedaan dalam mean.

```
>median(cs[2]) - median(cs[1])
```

1.5

Koefisien korelasi menunjukkan korelasi positif.

```
>correl(cs[1], cs[2])
```

0.7588307236

Korelasi peringkat adalah ukuran untuk urutan yang sama dalam kedua vektor. Korelasi ini juga cukup positif.

```
>rankcorrel(cs[1], cs[2])
```

0.758925292358

Membuat Fungsi baru

Tentu saja, bahasa EMT dapat digunakan untuk memprogram fungsi baru. Misalnya, kita mendefinisikan fungsi kemiringan.

$$sk(x) = \frac{\sqrt{n} \sum_i (x_i - m)^3}{(\sum_i (x_i - m)^2)^{3/2}}$$

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

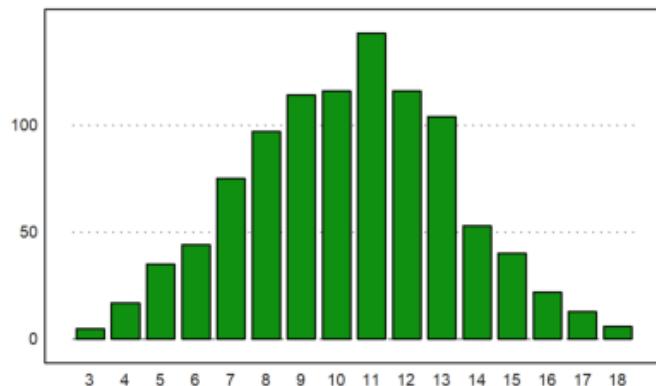
Euler dapat digunakan untuk mensimulasikan kejadian acak. Kita telah melihat contoh sederhana di atas. Berikut ini adalah contoh lainnya, yang mensimulasikan 1000 kali pelemparan 3 dadu, dan menanyakan distribusi dari jumlah tersebut.

```
>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]
```

We can plot this now.

```
>columnsplot(fs,lab=3:18):
```



Untuk menentukan distribusi yang diharapkan tidaklah mudah. Kami menggunakan rekursi tingkat lanjut untuk hal ini.

Fungsi berikut ini menghitung jumlah cara angka k dapat direpresentasikan sebagai jumlah n angka dalam rentang 1 hingga m. Fungsi ini bekerja secara rekursif dengan cara yang jelas.

```
>function map countways (k; n, m) ...
```

```

if n==1 then return k>=1 && k<=m
else
  sum=0;
  loop 1 to m; sum=sum+countways(k-#,n-1,m); end;
  return sum;
end;
endfunction

```

Berikut ini adalah hasil dari tiga lemparan dadu.

```
>countways(5:25,5,5)
```

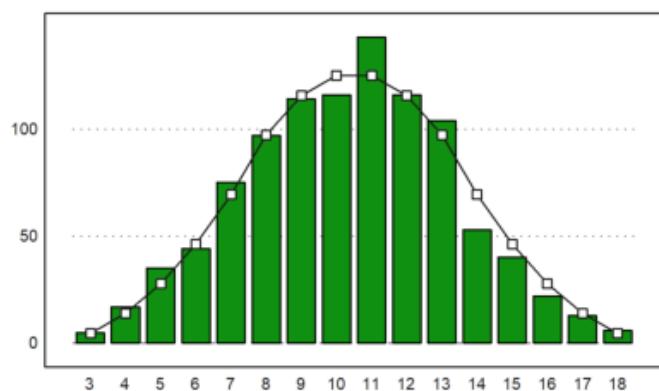
```
[1, 5, 15, 35, 70, 121, 185, 255, 320, 365, 381, 365, 320,
255, 185, 121, 70, 35, 15, 5, 1]
```

```
>cw=countways(3:18,3,6)
```

```
[1, 3, 6, 10, 15, 21, 25, 27, 27, 25, 21, 15, 10, 6, 3,
1]
```

Kami menambahkan nilai yang diharapkan ke plot.

```
>plot2d(cw/6^3*1000,>add); plot2d(cw/6^3*1000,>points,>add):
```



Untuk simulasi lainnya, deviasi nilai rata-rata dari n variabel acak berdistribusi normal 0-1 adalah $1/\sqrt{n}$.

```
>longformat; 1/sqrt(10)
```

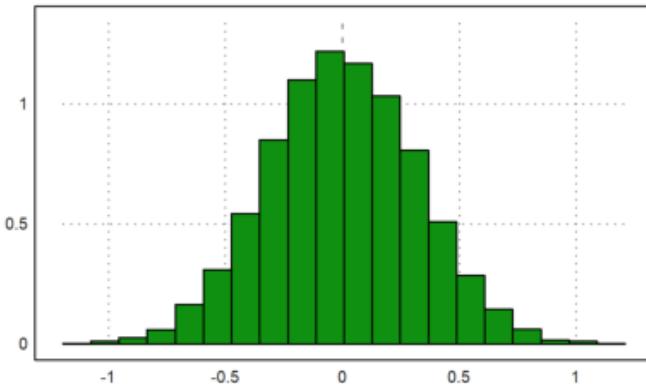
```
0.316227766017
```

Mari kita periksa dengan sebuah simulasi. Kami menghasilkan 10.000 kali 10 vektor acak.

```
>M=normal(10000,10); dev(mean(M)')
```

```
0.319493614817
```

```
>plot2d(mean(M)',>distribution):
```



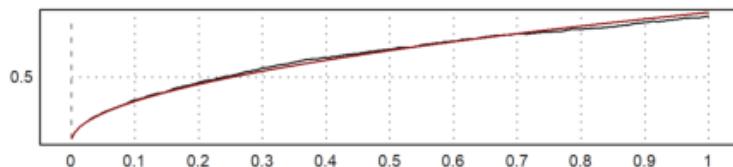
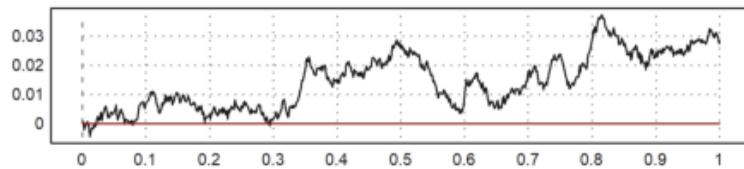
Median dari 10 bilangan acak berdistribusi normal 0-1 memiliki deviasi yang lebih besar.

```
>dev(median(M)')
```

0.374460271535

Karena kita dapat dengan mudah menghasilkan jalan acak, kita dapat mensimulasikan proses Wiener. Kami mengambil 1000 langkah dari 1000 proses. Kami kemudian memplot deviasi standar dan rata-rata dari langkah ke-n dari proses-proses ini bersama dengan nilai yang diharapkan dalam warna merah.

```
>n=1000; m=1000; M=cumsum(normal(n,m)/sqrt(m)); ...
>t=(1:n)/n; figure(2,1); ...
>figure(1); plot2d(t,mean(M)'); plot2d(t,0,color=red,>add); ...
>figure(2); plot2d(t,dev(M)'); plot2d(t,sqrt(t),color=red,>add); ...
>figure(0):
```



Tes

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 bisa 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

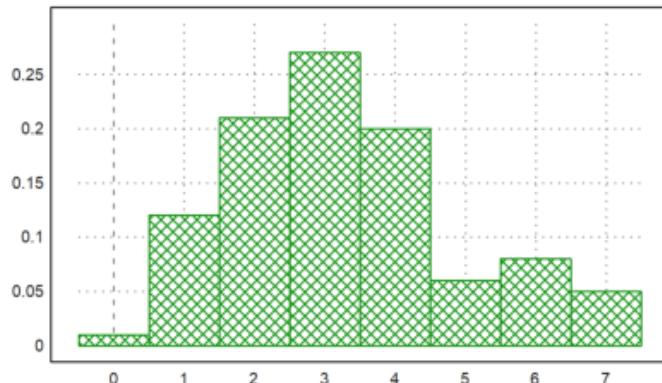
Pada contoh berikut, kita membuat 20 lemparan dadu secara acak sebanyak 100 kali dan menghitung jumlah dadu yang muncul. Rata-rata harus ada $20/6 = 3,3$ mata dadu.

```
>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="\\"/"):
```



```
>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 chi² 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=c("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].

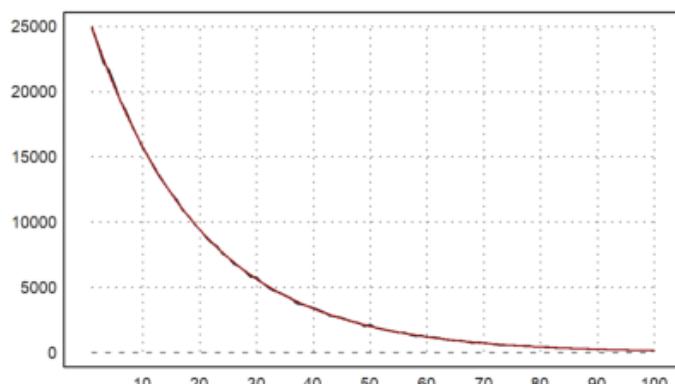
```
>n:=10000000; r:=random(1,n);
```

Selanjutnya, kami menghitung jarak antara dua angka yang kurang dari 0,05.

```
>a:=0.05; d:=differences(nonzeros(r<a));
```

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 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, %*%
```

```
[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   Grothmann"
```

© Ren   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 meng-ekstrak elemen-elemen yang bukan nol terlebih dahulu di EMT.

```
>x, x>5, x[nonzeros(x>5)]
```

```
[10.4, 5.6, 3.1, 6.4, 21.7]
[1, 1, 0, 1, 1]
[10.4, 5.6, 6.4, 21.7]
```

Sama seperti di R, vektor indeks dapat berisi pengulangan.

```
>x[[1,2,2,1]]
```

```
[10.4, 5.6, 5.6, 10.4]
```

Namun pemberian nama untuk indeks tidak dimungkinkan dalam EMT. Untuk paket statistik, hal ini mungkin sering diperlukan untuk memudahkan akses ke elemen-elemen vektor.

Untuk meniru perilaku ini, kita dapat mendefinisikan sebuah fungsi sebagai berikut.

```
>function sel (v,i,s) := v[indexof(s,i)]; ...
>s=["first","second","third","fourth"]; sel(x,[ "first","third"],s)
```

```
Trying to overwrite protected function sel!
Error in:
function sel (v,i,s) := v[indexof(s,i)]; ... ...
^
[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;
```

Untuk mengubah jenis tanggal di EMT, Anda dapat menggunakan fungsi seperti `complex()`.

```
>complex(1:4)
```

```
[ 1+0i ,  2+0i ,  3+0i ,  4+0i ]
```

Konversi ke string hanya dapat dilakukan untuk tipe data dasar. Format saat ini digunakan untuk penggabungan string sederhana. Tetapi ada fungsi-fungsi seperti `print()` atau `frac()`.

Untuk vektor, Anda dapat dengan mudah menulis fungsi Anda sendiri.

```
>function tostr (v) ...
```

```
s="[";  
loop 1 to length(v);  
  s=s+print(v[#],2,0);  
  if #<length(v) then s=s+","; endif;  
end;  
return s+"]";  
endfunction
```

```
>tostr(linspace(0,1,10))
```

```
[0.00,0.10,0.20,0.30,0.40,0.50,0.60,0.70,0.80,0.90,1.00]
```

Untuk komunikasi dengan Maxima, ada sebuah fungsi `convertmxm()`, yang juga dapat digunakan untuk memformat vektor untuk output.

```
>convertm xm(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 `tapply()` 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, 57.333333333, 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, 57.333333333]
```

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 data ini 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)'
```

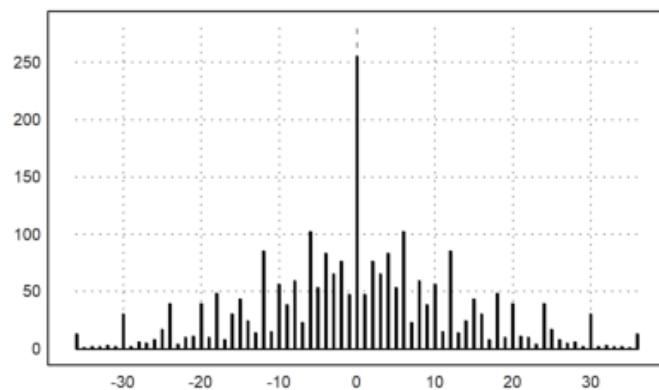
1	2	3	4	5
2	4	6	8	10
3	6	9	12	15
4	8	12	16	20
5	10	15	20	25

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"):
```



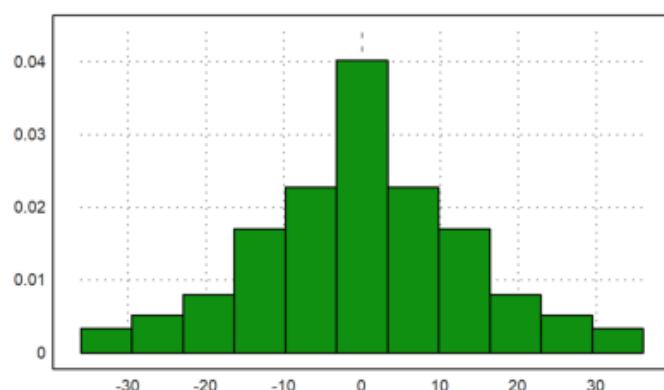
Selain kelipatan yang tepat, EMT dapat menghitung frekuensi dalam vektor.

```
>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.

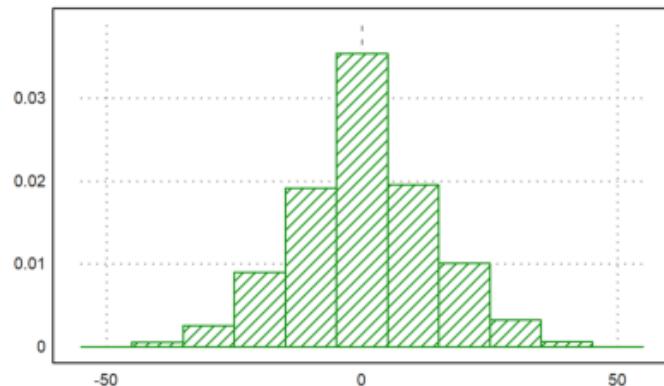
```
>plot2d(q,distribution=11):
```



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="/"):
```



Daftar

EMT memiliki dua jenis daftar. Yang 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.

Sebagai contoh, jika Anda memiliki sistem bahasa Indonesia (titik desimal dengan koma), Excel Anda membutuhkan nilai dengan koma desimal yang dipisahkan oleh titik koma dalam file csv (defaultnya adalah nilai yang dipisahkan dengan koma). File "test.csv" berikut ini akan muncul di folder current Anda.

```
>filename="test.csv"; ...  
>writematrix(random(5,3),file=filename,separator=",");
```

Anda sekarang dapat membuka file ini dengan Excel Indonesia secara langsung.

```
>fileremove(filename);
```

Terkadang kita memiliki string dengan token seperti berikut ini.

```
>s1:="f m m f m m m f f f m m f"; ...  
>s2:="f f f m m f f";
```

Untuk menandai ini, kita mendefinisikan vektor token.

```
>tok := [ "f", "m" ]
```

f

m

Kemudian kita dapat menghitung berapa kali setiap token muncul dalam string, dan memasukkan hasilnya ke dalam tabel.

```
>M:=getmultiplicities(tok,strtokens(s1))_ ...
>  getmultiplicities(tok,strtokens(s2));
```

Tulis tabel dengan tajuk token.

```
>writetable(M,labc=tok,labr=1:2,wc=8)
```

	f	m
1	6	7
2	5	2

Untuk statika, EMT dapat membaca dan menulis tabel.

```
>file="test.dat"; open(file,"w"); ...
>writeln("A,B,C"); writematrix(random(3,3)); ...
>close();
```

File terlihat seperti ini.

```
>printfile(file)
```

A,B,C
0.7003664386138074,0.1875530821001213,0.3262339279660414
0.5926249243193858,0.1522927283984059,0.368140583062521
0.8065535209872989,0.7265910840408142,0.7332619844597152

Fungsi readtable() dalam bentuknya yang paling sederhana dapat membaca ini dan mengembalikan sebuah koleksi nilai dan baris judul.

```
>L=readtable(file,>list);
```

Koleksi ini dapat dicetak dengan writetable() ke buku catatan, atau ke sebuah file.

```
>writetable(L,wc=10,dc=5)
```

A	B	C
0.70037	0.18755	0.32623
0.59262	0.15229	0.36814
0.80655	0.72659	0.73326

Matriks nilai adalah elemen pertama dari L. Perhatikan bahwa mean() dalam EMT menghitung nilai rata-rata dari baris-baris matriks.

```
>mean(L[1])
```

```
0.40472  
0.37102  
0.75547
```

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
```

CSV ini dapat dibuka di sistem bahasa Inggris ke Excel dengan klik dua kali. Jika Anda mendapatkan file seperti itu pada sistem Jerman, Anda perlu mengimpor data ke Excel dengan memperhatikan titik desimal. Namun, titik desimal juga merupakan format default untuk EMT. Anda dapat membaca sebuah matriks dari sebuah file dengan readmatrix().

```
>readmatrix(file)
```

```
0.82212 0.82153 0.77712  
0.84829 0.32378 0.65014  
0.14823 0.32975 0.62619
```

Dimungkinkan untuk menulis beberapa matriks ke dalam satu file. Perintah open() dapat membuka file untuk menulis dengan parameter "w". Standarnya adalah "r" untuk membaca.

```
>open(file,"w"); writematrix(M); writematrix(M'); close();
```

Matriks-matriks tersebut dipisahkan oleh sebuah baris kosong. Untuk membaca matriks, buka file dan panggil readmatrix() beberapa kali.

```
>open(file); A=readmatrix(); B=readmatrix(); A==B, close();
```

```
1 0 0  
0 1 0  
0 0 1
```

Di Excel atau spreadsheet serupa, Anda dapat mengekspor matriks sebagai CSV (nilai yang dipisahkan dengan koma). Pada Excel 2007, gunakan "save as" dan "format lain", lalu pilih "CSV". Pastikan, tabel saat ini hanya berisi data yang ingin Anda ekspor.

Berikut ini adalah contohnya.

```
>printfile("excel-data.csv")
```

```
0;1000;1000
1;1051,271096;1072,508181
2;1105,170918;1150,273799
3;1161,834243;1233,67806
4;1221,402758;1323,129812
5;1284,025417;1419,067549
6;1349,858808;1521,961556
7;1419,067549;1632,31622
8;1491,824698;1750,6725
9;1568,312185;1877,610579
10;1648,721271;2013,752707
```

Seperti yang Anda lihat, sistem Jerman saya menggunakan titik koma sebagai pemisah dan koma desimal. Anda dapat mengubahnya di pengaturan sistem atau di Excel, tetapi tidak perlu untuk membaca matriks ke dalam EMT.

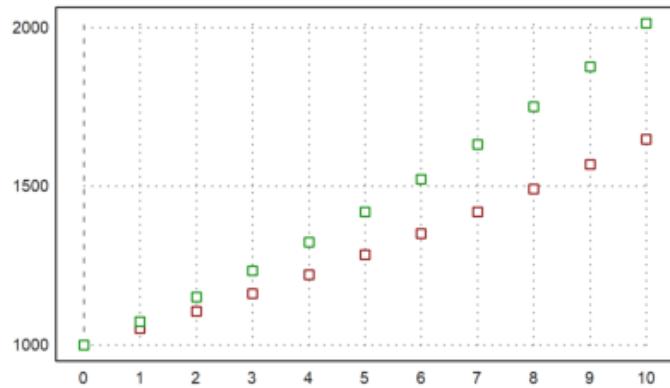
Cara termudah untuk membaca ini ke dalam Euler adalah readmatrix(). Semua koma digantikan oleh titik dengan parameter >comma. Untuk CSV bahasa Inggris, hilangkan saja parameter ini.

```
>M=readmatrix("excel-data.csv",>comma)
```

0	1000	1000
1	1051.3	1072.5
2	1105.2	1150.3
3	1161.8	1233.7
4	1221.4	1323.1
5	1284	1419.1
6	1349.9	1522
7	1419.1	1632.3
8	1491.8	1750.7
9	1568.3	1877.6
10	1648.7	2013.8

Let us plot this.

```
>plot2d(M' [1],M' [2:3],>points,color=[red,green]'):
```



Ada beberapa cara yang lebih mendasar untuk membaca data dari file. Anda dapat membuka file dan membaca angka baris demi baris. Fungsi `getvectorline()` akan membaca angka dari sebuah baris data. Secara default, fungsi ini mengharapkan sebuah titik desimal. Tetapi fungsi ini juga dapat menggunakan koma desimal, jika Anda memanggil `setdecimaldot(",")` sebelum menggunakan fungsi ini.

Fungsi berikut ini adalah contohnya. Fungsi ini akan berhenti pada akhir file atau baris kosong.

```
>function myload (file) ...
```

```
open(file);
M=[];
repeat
    until eof();
    v=getvectorline(3);
    if length(v)>0 then M=M_v; else break; endif;
end;
return M;
close(file);
endfunction
```

```
>myload(file)
```

```
0.82212  0.82153  0.77712
0.84829  0.32378  0.65014
0.14823  0.32975  0.62619
```

Anda juga dapat membaca semua angka di 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>"
```

```
<tr><td>1145.45</td><td>5.6</td><td>-4.5</td><tr>
```

Untuk mengekstrak ini, kita menggunakan ekspresi reguler, yang mencari

- tanda kurung tutup >,
- setiap string yang tidak mengandung tanda kurung dengan

sub-pencocokan "(...)".

- kurung pembuka dan kurung penutup menggunakan solusi terpendek,
- sekali lagi, semua string yang tidak mengandung tanda kurung,
- dan sebuah kurung pembuka <.

Ekspresi reguler agak sulit untuk dipelajari tetapi sangat kuat.

```
>{pos,s,vt}=strxfind(line,>([<>]+)<.+?>([<>]+)<" );
```

Hasilnya adalah posisi kecocokan, string yang cocok, dan vektor string untuk sub-cocokan.

```
>for k=1:length(vt); vt[k](), end;
```

```
1145.5
5.6
```

Berikut ini adalah fungsi yang membaca semua item numerik antara <td> dan </td>.

```
>function readtd (line) ...
```

```
v=[]; cp=0;
repeat
  {pos,s,vt}=strxfind(line,"<td.*?>(.+?)</td>",cp);
  until pos==0;
  if length(vt)>0 then v=v|vt[1]; endif;
  cp=cp+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 2022-05-18

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]
```