

# Tugas LaTeX dan Markdown

Lasigi Yatindra Jago lasigiyatindra.2023

November 2024

## 1 Pendahuluan

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 Tutorias and Example", lalu pilih file "00 First Steps.en". Perhatikan, file notebook EMT memiliki ekstensi ".en". Melalui notebook ini Anda akan belajar menggunakan software Euler untuk menyelesaikan berbagai masalah matematika.

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

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

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

Seperti yang Anda baca, notebook ini berisi tulisan (teks) berwarna hijau, yang dapat Anda edit dengan mengklik kanan teks atau tekan menu Edit -gt;

Edit Comment atau tekan [F5], dan juga baris perintah EMT yang ditandai dengan "gt;" 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* :  $\text{integrate}(x^3, x) = \text{integrate}(x^3, x) + C$  - *http* : <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

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

maxima:  $\text{integrate}(x^3, x) = \text{integrate}(x^3, x) + C$

<http://www.euler-math-toolbox.de> | <http://www.euler-math-toolbox.de>

<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.

JAWABAN

NAMA : LASIGI YATINDRA JAGO

KELAS : MATEMATIKA B 2023

NIM : 23030630008

$\tan(90^\circ)$

$1.63312393532e+16$

$1+5/8+3/4+7/9+3$

$6.15277777778$

$(9^5 + 47)/(8^4 + 715)$

$14.0626041419$

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 -.

\* *^* lebih kuat dari \*, sehingga  $r^2$  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.

$\sin(45^\circ)$ ,  $\cos(\pi)$ ,  $\log(\sqrt{E})$

$0.707106781187 - 1$   $0.5$

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 -gt; Reference)

Untuk membuat rangkaian komputasi lebih mudah, Anda dapat merujuk ke hasil sebelumnya dengan  $\%$  perhitungan dalam baris perintah yang sama.

$(\sqrt{5}+1)/2,$

$1.61803398875$   $2$

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.

JAWABAN

NAMA : LASIGI YATINDRA JAGO

KELAS : MATEMATIKA B 2023

NIM : 23030630008

$\sin(30^\circ)/(\sqrt{9}) + \cos(45^\circ)$  ,  $\cos(30^\circ)/(\sqrt{16})$

0.873773447853 0.216506350946

$\cos(45^\circ)/(\sqrt{4}) + \sin(90^\circ)$  ,  $\sin(45^\circ)/(\sqrt{4})$

1.35355339059 0.353553390593

—  
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 bolehtidak perluditulisdengan mengaktifkan easy units.*

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 -gt;.

```

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 mengatrtampilan 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 12001.0310, longE, longestpi
1612.7 2.71828182846 3.141592653589793

```

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

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

```

longestformat; pi, defformat; pi
3.141592653589793 3.14159265359

```

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

```

1/7+1/4, fraction
0.392857142857 11/28

```

Perintah Multibaris

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

belumnya. 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  $ax^2 + bx + c = 0$ secaramanual... D =  
sqrt(b2/(a24) - c/a); ... - b/(2a) + D, ... - b/(2a) - 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  
km1000cm 0.01 mm0.001nm 1853.24496 gram0.001m 1 hquantum6.62606957e-  
34atm 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 "inrandom" berikut ini!

```
inrandom(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(w2)
0.35
```

#### Bilangan Kompleks

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

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 imajiner 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 imajiner x terlalu besar, hasilnya akan menampilkan pesan kesalahan.

```
gt;sqrt(-1) // Error! gt;sqrt(complex(-1))
z=2+3I, re(z), im(z), conj(z), arg(z), deg(arg(z)), deg(arctan(3/2))
2+3i 2 3 2-3i 0.982793723247 56.309932474 56.309932474
deg(arg(I)) // 90°
90
sqrt(-1)
Floating point error! Error in sqrt Error in: sqrt(-1) ...
sqrt(complex(-1))
```

0+1i

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

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

```
complex(-1), sqrt(-1+0i 0+1i
```

Matematika Simbolik

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

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

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

```
(a+b)2
2 (b + a)
expand((a+b)2), factor(x2 + 5x + 6)
2 2 b + 2 a b + a
(x + 2) (x + 3)
solve(ax2 + bx + c, x)//rumusabc
2 2 (- sqrt(b - 4 a c)) - b sqrt(b - 4 a c) - b [x = —————, x =
—————] 2 a 2 a
(a2 - b2)/(a + b), ratsimp(
2 2 a - b ———- b + a
a - b
10! // nilai faktorial (modus EMT)
3628800
10! //nilai faktorial (simbolik dengan Maxima)
3628800
```

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

```
factor(1000) // mencari semua faktor 1000 (EMT)
[2, 2, 2, 5, 5, 5]
:: factor(1000) // faktorisasi prima 1000 (dengan Maxima)
3 3 2 5
:: factor(20!)
18 8 4 2 2 3 5 7 11 13 17 19
```



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

```
::: binomial(5,2); // nilai C(5,2)
10
::: binomial(m,4); // C(m,4)=m!/(4!(m-4)!)
(m - 3) (m - 2) (m - 1) m ----- 24
::: trigexpand(cos(x+y)); // rumus cos(x+y)=cos(x) cos(y)-sin(x)sin(y)
cos(x) cos(y) - sin(x) sin(y)
::: 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)//menyederhanakanfungsi trigonometri
4 sin(x) + cos(x) ----- 3 cos(x)
```

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

```
p1 = (x^3 + 1)/(x + 1)
3 x + 1 ----- x + 1
ratsimp(p1)
2 x - x + 1
```

Untuk mensubstitusikan suatu nilai ke dalam variabel dapat digunakan perintah "with".

```
p1 with x=3 // (3^3 + 1)/(3 + 1)
7
p1 with x=a+b, ratsimp(
3 (b + a) + 1 ----- b + a + 1
2 2 b + (2 a - 1) b + a - a + 1
diff(p1,x) //turunan p1 terhadap x
2 3 3 x x + 1 ----- x + 1 2 (x + 1)
integrate(p1,x) // integral p1 terhadap x
3 2 2 x - 3 x + 6 x ----- 6
```

Tampilan Matematika Simbolik dengan LaTeX

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

Perhatikan, hal ini hanya dapat menghasilkan tampilan yang diinginkan apabila komputer Anda sudah terpasang software LaTeX.

```
(a + b)^2
expand((a + b)^2), factor(x^2 + 5x + 6)
solve(ax^2 + bx + c, x) // rumus abc
(a^2 - b^2)/(a + b), ratsimp(
```

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.

JAWABAN

NAMA : LASIGI YATINDRA JAGO

KELAS : MATEMATIKA B 2023

NIM : 23030630008

function f(x) = (cos(x)sin(y)<sup>2</sup>)/2

2 cos(x) sin (y) ————— 2

function f(x) = integrate(f(x),y)

sin(2 y) cos(x) (y - ———) 2 ————— 4

## 2 Aljabar

Tugas Individu Nama : Lasigi Yatindra Jago

Kelas : MATEMATIKA B 2023

NIM : 23030630008

EMT untuk Perhitungan Aljabar

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

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

Contoh pertama

Menyederhanakan bentuk aljabar:

$$5x^{-4}y^3 \times -8x^5y^{-6}$$

$$5x(-4)y^3 - 8x^5y(-6)$$

Menyederhanakan fungsi :

$$5y^2 + 3x^6 - 7y^2 + 2x^2$$

$$5y^2 + 3x^6 - 7y^2 + 2x^2$$

Menjabarkan:

$$(5x^{-4} + y^3)(-8x^5 - y^{-6})$$

*showev('expand((5x<sup>-4</sup> + y<sup>3</sup>)(-8x<sup>5</sup> - y<sup>-6</sup>)))*

$$\text{expand}\left(\left(y^3 + 5x^{\{-4\}}\right)\left(-y^{\{-6\}} - 8x^5\right)\right) = -y^{\{-6\}+3} - 5x^{\{-4\}}y^{\{-6\}} - 8x^5y^3 - 40x^{\{-4\}+5}$$

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:=3; h:=4; pi*r2*h/3
37.6991118431
```

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

```
pi*2rh,
75.3982236862 150.796447372
```

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

```
x := 4;
x := cos(x) // nilai cosinus (x dalam radian)
-0.653643620864
x := cos(x)
0.793873449226
```

Jika dua garis terhubung dengan "..." kedua garis akan selalu dieksekusi secara bersamaan.

```
x := 3.5; ... x := (x+2/x)/2, x := (x+2/x)/2, x := (x+2/x)/2,
2.03571428571 1.50908521303 1.41719571011
```

Ini juga merupakan cara yang baik untuk menyebarkan long command 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.

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

```
// This line will not be visible once the cursor is off the line
Garis yang diawali dengan
81
```

Euler Math Toolbox 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=6; for i=1 to 10; x := (x+2/x)/2, end; // menghitung akar 2
```

```
3.16666666667 1.89912280702 1.47612029496 1.4155117098 1.41421415763
1.41421356237 1.41421356237 1.41421356237 1.41421356237 1.41421356237
```

Tidak apa-apa untuk menggunakan multi-line. Pastikan baris diakhiri dengan "...".

```
x := 2.5; // 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 Epi piE; then "HaloRakyatku!", endif;
```

Halo Rakyatku!

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. Dan 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(45°)/cos(60°))
```

```
1.189207115
```

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(5.7))
```

```
5.7
```

Sintaks Dasar

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

Untuk menyetel variabel, gunakan "=" atau ":=". Demi kejelasan, pengantar ini menggunakan bentuk yang terakhir/terbaru. Spasi tidak menjadi masalah. Tetapi ruang antara perintah diharapkan untuk ada.

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

```
g:=10.73; t:=4.2; 1/2gt2
```

```
94.6386
```

EMT menggunakan sintaks pemrograman untuk ekspresi. Untuk mengetik

$$e^2 \cdot \left( \frac{1}{9 + 5 \log(0.7)} + \frac{4}{9} \right)$$

Anda harus mengatur tanda kurung dengan 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/(9 + 5log(0.7)) + 4/9)
4.30791848586
```

Untuk menghitung ekspresi rumit seperti

$$\left( \frac{\frac{5}{8} + \frac{7}{6} + 3}{\frac{3}{7} + \frac{5}{9}} \right)^2 \pi$$

Anda harus memasukkannya dalam bentuk baris.

```
((5/8 + 7/6 + 3) / (3/7 + 5/9))^2pi
74.4767625442
```

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.

```
2/3+5/7, fraction
1.38095238095 29/21
```

Perintah Euler dapat berupa ekspresi atau perintah primitif. Ekspresi terbuat dari operator dan fungsi. Jika diperlukan, hal tersebut harus berisi tanda kurung untuk memaksa urutan eksekusi yang benar. Jika ragu, memasang braket atau tanda kurung adalah ide yang bagus. Perhatikan bahwa EMT menunjukkan tanda kurung buka dan tutup saat mengedit baris perintah.

```
(cos(pi/4)+2)^3(sin(pi/4) + 5)^2
646.172032434
```

Operator numerik Euler meliputi

+ unary atau operator plus - unary atau operator minus \* operator perkalian / operator pecahan . produk matriks *a<sup>b</sup>dayauntukpositifaataubilanganbulatb(a\*  
\*bjugaberfungsi)n!operatorfaktorial*  
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. ln untuk log.

```
ln(E^4), arctan(tan(0.75)), logbase(30, 10)
```



bekerja untuk ekspresi juga (lihat dibawah).

```
"1234.5567"()
```

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

```
v:= ["Indonesia","Malaysia","Brunei Darussalam"]
```

Indonesia Malaysia Brunei Darussalam

Vektor pada string kosong dilambangkan dengan [none]. Dan vektor string dapat digabungkan dengan '—'.

```
w:= [none] ; w—v—v
```

Indonesia Malaysia Brunei Darussalam Indonesia Malaysia Brunei Darussalam

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"beta; = " + 90 + u"deg; " // pdfLaTeX mungkin gagal menampilkan
```

secara benar

```
= 90°
```

Dalam komentar, entitas yang sama seperti alpha; beta; dll dapat digunakan untuk latexs.

Ada beberapa fungsi untuk membuat atau menganalisis string unicode.

Fungsi strtchr() 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]
```

Perintah ini menghasilkan array atau daftar angka berupa vektor angka yang mewakili karakter dalam string dalam bentuk kode Unicode.

Fungsi kebalikannya adalah chartoutf().

```
v[1]=strtochar(u"Auml;")[1]; chartoutf(v)
```

Ä is a German letter

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

```
a="We have alpha;=beta;."; utf(a)// PdfLaTeX mungkin gagal menampilkan
```

nya

```
We have =.
```

Memungkinkan juga untuk menggunakan entitas numerik.

```
u"196;lphabet"
```

Älphabet

Nilai Boolean

Nilai boolean direpresentasikan dengan 1=true atau 0=false dalam euler.

String dapat dibandingkan, seperti halnya angka.

```
"saya" "aku", 6==3
```

```
1 0
```

```
5 1, "mobil"=="motor"
```

```
1 0
```

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

```
2:E — E:3
```

```
1
```

```
6 E E:2
```

```
0
```

Operator Boolean mematuhi aturan bahasa matriks

```
(2:9) 3, nonzeros (
```

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

Kita dapat menggunakan fungsi bukan nol() untuk mengekstrak elemen tertentu dari vektor. Dalam contoh, menggunakan isprime bersyarat(n).

```
N=5 — 7:2:50 // N berisi elemen 5 dan bilangan bilangan ganjil dari 7:50
```

```
[5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49]
```

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

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

Output Formats

Default output formats EMT adalah 12 digit. Untuk memastikan yang kita lihat adalah bentuk default, maka perlu direset format.

```
defformat; pi
```

```
3.14159265359
```

Secara internal, EMT menggunakan standar IEEE (Institute of Electrical and Electronics Engineers) untuk bilangan ganda dengan sekitar 16 digit desimal. Untuk melihat bentuk digit penuh, gunakan perintah "longestformat" atau gunakan operator "longest" untuk memunculkannya.

```
longest pi
```

```
3.141592653589793
```

```
longestformat; pi
```

```
3.141592653589793
```

Berikut ini adalah representasi heksadesimal internal dari bilangan ganda.

```
printhe(pi)
```

```
3.243F6A8885A30*160
```

Heksadesimal adalah sistem bilangan yang menggunakan basis 16. Di mana angka 0 hingga 9 (untuk mewakili nilai 0 hingga 9) dan huruf A hingga F (untuk mewakili nilai 10 hingga 15).

Format standarnya adalah 12.

```
format(12); 1/7
```

```
0.142857142857
```

Format output dapat diubah secara permanen dengan perintah format.

```
format(12,5); 1/9, pi, cos(1)
```

```
0.11111 3.14159 0.54030
```

Format standar untuk skalar adalah 12, tetapi ini dapat diubah.

```
setscalarformat(7); pi
```

```
3.14159
```

Begitu juga dengan fungsi "longestformat" mengatur format skalar.



```

longestformat; pi
3.141592653589793
Notes: beberapa format output yang penting.
shortestformat shortformat longformat, longestformat format(length,digits)
goodformat(length) fracformat(length) deformat
Akurasi internal EMT adalah sekitar 16 digit desimal mengikuti standar
dari IEEE. Angka disimpan dalam format internal. Namun, format output
EMT dapat diatur secara fleksibel.
fraction 2+3/10+7/14+21/7
29/5
fraction 57/2/32/5
7/3
Digunakan untuk menampilkan ke bentuk pecahan sederhana.
longest 0.2+0.25+0.3+0.3+0.25+0.2+0.5-2.6
-0.6000000000000001
Perintah ini menunjukkan presisi penuh dari operasi aritmetika yang meli-
batkan angka-angka kecil, dan bagaimana kesalahan akumulatif bisa muncul
dalam perhitungan biner.
Expressions
String atau nama dapat digunakan untuk menyimpan ekspresi matematika,
yang dapat dievaluasi oleh EMT. Gunakan tanda kurung setelah ekspresi.
k:=5; fx:="pi"; fx()
15.70796326794897
Ekspresi akan selalu menggunakan global variable, bahkan jika ada variabel
dalam fungsi dengan nama yang sama.
fx:="acos(x)"; fx(10,a=0.8)
-0.671257223261162
Menggunakan parameter yang ditetapkan ke x, y, z, dll. Jika tidak, eval-
uasi ekspresi dalam fungsi dapat memberikan hasil yang membingungkan bagi
pengguna yang memanggil fungsi tersebut.
at:=6; function f(expr,x,at) := expr(x); ... f("atx^4",2,3)//computes6 ×
2^4not32^4
96
Menggunakan global variable pada fungsi, dimana "at" merupakan global
variables. Jika ingin menggunakan nilai lain untuk "at" perlu menambahkan
"at=vaue".
ut:=5; function f(expr,x,p) := expr(x,ut=p); ... f("utx^3",4,2)//computes2 ×
4^3not54^3
128
Walaupun "ut" sebagai global variable sudah didefinisikan, tetapi didefin-
isikan kembali pada ekspresi fungsinya dimana "ut=p" sehingga nilainya berganti
dari yang awalnya ut=5 menjadi ut=p=2.
f = x^2
2 x
function f(x) := x^4
f(2)

```

16

Ekspresi dalam x sering digunakan seperti fungsi.

Mendefinisikan fungsi dengan nama yang sama seperti ekspresi simbolik global (f amp;=) menghapus nilai variabel sebelumnya untuk menghindari kebingungan antara ekspresi simbolik dan fungsi.

"@(a,b)  $a^3 + b^2$ ",

@(a,b)  $a^3 + b^2$

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

fx = 2x-3t; ... t=2.5; fx(0.8)

-5.9

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

fx(1,t=1.5)

-2.5

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

Symbolic Mathematics

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

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

35!

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

Mari kita hitung!

ateks: C(35,15)= $35! / (20! \cdot 15!)$

$35! / (20! \cdot 15!) // \text{nilai } C(35, 15)$

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

*binomial(35,15) // menghitung C(35,15) menggunakan fungsi binomial()*

Untuk mempelajari lebih lanjut tentang fungsi tertentu klik dua kali di atasnya. Misalnya, coba klik dua kali pada "amp;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.

*binomial(x,3) with x = 5 // substitusix = 5 ke C(x,3)*

Dengan begitu kita dapat menggunakan solusi persamaan dalam persamaan lain.

*sol = solve(x<sup>2</sup> + 3x = 9, x); sol, sol(), float(sol)*

*[-4.854101966249685, 1.854101966249685]*

Untuk mencetak ekspresi simbolis dengan LaTeX, gunakan *didepan amp; (ataudapat menghilangkan amp;)* sebelum jika tidak menginstal LaTeX.

Untuk mendapatkan solusi simbolis tertentu, seseorang dapat menggunakan "with" dan index.

`solve(x2 + 3x = 1, x), x2 = x with`

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

`sol = solve([x+y=5, x2 + y2 = 15], [x, y]); sol, xy with sol[1]`

Eksprei 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)").

Functions

Dalam matematika, fungsi aljabar adalah fungsi yang bisa didefinisikan sebagai akar dari sebuah persamaan aljabar. Fungsi aljabar merupakan ekspresi aljabar menggunakan sejumlah suku terbatas, yang melibatkan operasi aljabar seperti penambahan, pengurangan, perkalian, pembagian, dan peningkatan menjadi pangkat pecahan.

contohnya:

$f(x) = 1/x$

$f(x) = \text{sqrt}(x)$

$f(x) = (\text{sqrt}(1 + x^3)) / (x^{3/7} - (\text{sqrt}(7)x^{1/3}))$

di EMT fungsi adalah program yang didefinisikan dengan perintah "function". Fungsi dapat menjadi fungsi satu baris atau fungsi multi baris. Fungsi satu baris dapat berupa numerik atau simbolis. Fungsi satu baris didefinisikan oleh " := ".

`function f(x):=xsqrt(x2 + 1)`

Lalu, semua fungsi pasti dapat didefinisikan oleh fungsi satu baris. Suatu fungsi dapat dievaluasi, contohnya kita akan mencari nilai f(5) dari fungsi f diatas.

`f(2)`

4.472135955

Fungsi ini dapat digunakan juga dalam vektor, dengan mengikuti aturan bahasa matrik Euler, karena ekspresi yang digunakan dalam fungsi divektorkan. Kita akan mencobanya menggunakan fungsi f di atas.

`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 juga dapat menjadi plot, hanya dengan 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 ingin menimpa fungsi bawaan bisa menambahkan kata kunci "overwrite". Menimpa fungsi bawaan bisa berbahaya dan menyebabkan masalah bagi fungsi lain yang bergantung pada fungsi tersebut.

Jika ingin memanggil fungsi bawaan bisa memakai "», jika fungsi tersebut merupakan fungsi inti Euler.

`function overwrite sin(x) := _sin(x°) // tentukan kembali sinus dalam derajat`

`sin(45)`

0.707106781187

Jika ingin menghapus definisi dari sin dan mendefinisikannya ulang, menggunakan perintah "forget"

```
forget sin; sin(pi/4)
0.707106781187
```

Default Parameters

Parameter default adalah fungsi parameter yang memiliki nilai awal.

Fungsi numerik dapat memiliki parameter default.

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

Menghilangkan parameter ini menggunakan nilai default.

```
f(4)
```

```
16
```

Menimpa default value.

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

```
80
```

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

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

```
16
```

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

```
function f(x):= ax^2
```

```
a=6; f(2)
```

```
24
```

Tetapi parameter yang ditetapkan menimpa global value.

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 "&=" ".FungsisimbolisdidefinisikandalamEulerdanmaxima,dan

```
function g(x) = x^3 - xexp(-x);g(x)
```

Fungsi simbolik dapat digunakan dalam ekspresi simbolik

```
diff(g(x), x),
```

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

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

```
178.635099908
```

Itu juga dapat digunakan untuk mendefinisikan fungsi atau ekspresi simbolik lainnya.

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

```
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

Dengan `amp`:= fungsinya simbolis, dan dapat digunakan dalam ekspresi simbolik lainnya. Contohnya dalam integral tak tentu sebagai berikut.

```
function P(x,n) = (2x-1)^n;P(x,n)
function Q(x,n) = (x+2)^n;Q(x,n)
```

```
P(x,4),expand(
P(3,4)
```

625

```
P(x,4) + Q(x,3),expand(
P(x,4) - Q(x,3),expand(
P(x,4)Q(x,3),expand(
P(x,4)/Q(x,1),expand(
function f(x) = x^3 - 2;f(x)
```

Dengan `amp`:= maka fungsi adalah simbolik, dan dapat digunakan di ekspresi simbolik lainnya.

```
integrate(f(x),x)
```

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

```
//latex: f(x) = \int_1^x t^t dt,
```

yang tidak dapat dinilai secara simbolis.

Jika kita mendefinisikan kembali fungsi dengan kata kunci `"map"` 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 menggunakan suatu parameter `"base"` maupun tidak.

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

Selain itu, dimungkinkan untuk menggunakan parameter yang ditetapkan.

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

2

Sering kali, kita ingin menggunakan fungsi untuk vektor di satu tempat, dan untuk elemen individual di tempat lain. Ini tepat terjadi dengan vektor parameter.

```
function f([a,b]) = a^2 + b^2 - ab + b;f(a,b), f(x,y)
```

Fungsi simbolik seperti itu dapat digunakan untuk variabel simbolik.

etapi fungsi ini juga dapat digunakan untuk vektor numerik.

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

17

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

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

diff(expr, y, 2) + diff(expr, x, 2)  
 $realpart((x + Iy)^4)$ ,lapl(

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

function f(x,y) = factor(lapl((x+y<sup>2</sup>)<sup>5</sup>, x, y));f(x,y)

Ringkasam:

- amp;= mendefinisikan fungsi simbolis,
- := mendefinisikan fungsi numerik,
- amp;amp;= 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<sup>2</sup> - 2", 1)  
 1.41421356237

Ini juga berfungsi untuk fungsi simbolik, perhatikan fungsi berikut ini.

$solve(x^2 = 2, x)$   
 $solve(x^2 - 2, x)$   
 $solve(ax^2 + bx + c = 0, x)$   
 $solve([ax + by = c, dx + ey = f], [x, y])$

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.

px = 4x<sup>8</sup> + x<sup>7</sup> - x<sup>4</sup> - x;px  
 solve(px,1,y=2), px(  
 0.966715594851 2

Memecahkan ekspresi simbolis dalam bentuk simbolis mengembalikan daftar solusi. Kami menggunakan

pemecah simbolik solve() yang disediakan oleh Maxima.

sol = solve(x<sup>2</sup> - x - 1, x);sol

Cara termudah untuk mendapatkan nilai numerik adalah dengan mengevaluasi solusi secara numerik seperti ekspresi.

longest sol()  
 -0.6180339887498949 1.618033988749895

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

$x^2$ withsol[1],expand(x<sup>2</sup> - x - 1withsol[2])

Memecahkan sistem persamaan secara simbolis dapat dilakukan dengan vektor persamaan dan solver simbolis solve(). Hasilnya dalam bentuk persamaan.

`solve([x + y = 2, x3 + 2y + x = 4], [x, y])`  
 Fungsi `f()` dapat melihat variabel global. Namun seringkali kita ingin menggunakan parameter lokal.

dengan `a=3`.

`function f(x,a) := xa + ax;`

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

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

`-0.710242150858`

Ini juga bekerja dengan ekspresi. Tapi daftar elemen yang ada harus digunakan.

`solve("x2 + ax", a = 3, 2, y = 0.1)`

`0.0329709716756`

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)`.

Eliminasi Fourier adalah analog dari eliminasi Gauss untuk linear (persamaan atau pertidaksamaan). Panggilan fungsi `'fourier_elim([eq1, eq2, ...], [var1, var2, ...])'` melakukan eliminasi

`load(fourier_elim)`

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

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

`fourier_elim([x2 - 1 < 0], [x])`

`fourier_elim([x2 - 4 < 0], [x])`

`fourier_elim([x2 - 9 < 0], [x])`

`fourier_elim([x < 10], [x])`

Ketika himpunan penyelesaiannya adalah kosong maka `'emptyset'`, dan ketika himpunan penyelesaiannya adalah semua bilangan real, maka `'universalset'`; sebagai contoh

`fourier_elim([min f < x, x < inf], [x])`

`fourier_elim([x < 1, x < 1], [x])`

Untuk persamaan nonlinier, `'fourier_elim'` mengembalikan sebuah daftar persamaan yang disederhanakan :

`fourier_elim([x3 - 8 < 0], [x])`

`fourier_elim([cos(x) < 1/2], [x])`

Alih-alih sebuah daftar pertidaksamaan, `'fourier_elim'` juga dapat berupa disjungsi atau konjungsi logika :

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

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

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

Fungsi `'fourier_elim'` mendukung operator pertidaksamaan `'lt', 'le', 'gt', 'ge'` ; `'and', 'or', 'xor', 'not'` ; dan `'='` .

Kode eliminasi Fourier memiliki sebuah preprocessor yang mengubah beberapa persamaan nonlinier yang melibatkan nilai absolut, minimum, dan fungsi maksimum menjadi linear dalam persamaan. Selain itu, preprocessor menangani beberapa ekspresi yang merupakan hasil kali atau hasil bagi dari suku-suku linier:

*fourier\_elim*([*max*(*x*, *y*) 6, *x*8, *abs*(*y* - 1) 12], [*x*, *y*])

*fourier\_elim*([(*x* + 2)/(*x* - 4) <= 2], [*x*])

Bahasa Matriks

Dalam matematika, matriks adalah susunan[1] bilangan, simbol, atau ekspresi yang disusun dalam baris dan kolom sehingga membentuk suatu bangun persegi

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

Matriks 1x2

a=[1;2]

1 2

b=[3,4;5,6]

3 4 5 6

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

1 2 3 4 5 6 7 8 9

Transpose matriks adalah matriks baru yang diperoleh dengan cara menukar elemen-elemen baris menjadi elemen kolom atau sebaliknya.

a'

[1, 2]

b'

3 5 4 6

c'

1 4 7 2 5 8 3 6 9

Invers matriks adalah matriks baru yang merupakan kebalikan dari matriks asal

inv(b)

-3 2 2.5 -1.5

Perkalian matriks sendiri adalah proses mengalikan setiap elemen baris pada matriks pertama dengan elemen kolom pada matriks kedua.

b.a

11 17

Perkalian dari matriks dengan invers matriks itu sendiri akan menghasilkan matriks identitas

b.inv(b)

1 0 0 1

Perkalian matriks dan perpangkatan matriks

b.b

29 36 45 56

b<sup>2</sup>

9 16 25 36

b.b.b

267 332 415 516

power(b,3)

267 332 415 516

Pembagian matriks

a/a



```

1 1
a/b
0.333333 0.25 0.4 0.333333
Perkalian invers matriks dengan matriks lainny
b
a
1 -0.5
inv(b).a
1 -0.5
Perkalian skalar
b2
6 8 10 12
2b
6 8 10 12
[1,2]2
[2, 4]
Fungsi Matriks Lainnya
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
A=random(3,4)
0.655416 0.200995 0.893622 0.281887 0.525 0.314127 0.444616 0.299474 0.28269
0.883227 0.270906 0.704419
A=1
0.655416 0.200995 0.893622 0.281887 1 0.525 0.314127 0.444616 0.299474 1
0.28269 0.883227 0.270906 0.704419 1
[v,v]
[1, 2, 3, 1, 2, 3]
[v;v]
1 2 3 1 2 3
[v',v']
1 1 2 2 3 3
"[x,x^2]"(v')
1 1 2 4 3 9
length(2:10)
9
ones(2,2)
1 1 1 1
zeros(2,2)
0 0 0 0
ones(5)6
[6, 6, 6, 6, 6]
random(1,2)
[0.217693, 0.445363]

```

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

```
redim(1:9,3,3)
1 2 3 4 5 6 7 8 9
function rep(v,n) := redim(dup(v,n),1,ncols(v))
rep(1:3,5)
[1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3]
multdup(1:3,3)
[1, 1, 1, 2, 2, 2, 3, 3, 3]
```

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

Keduanya tidak memiliki jumlah kolom yang sama, kolom yang lebih pendek akan diisi dengan 0.

```
flipx(1:5)
[5, 4, 3, 2, 1]
rotright(1:5)
[2, 3, 4, 5, 1]
rotright(1:5)
[5, 1, 2, 3, 4]
```

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]
```

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

```
A=id(3)
1 0 0 0 1 0 0 0 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(1:4)
[1, 1.41421, 1.73205, 2]
```

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

```
x=3:0.05:6; y=log(x)
[1.09861, 1.11514, 1.1314, 1.1474, 1.16315, 1.17865, 1.19392, 1.20896, 1.22378,
1.23837, 1.25276, 1.26695, 1.28093, 1.29473, 1.30833, 1.32176, 1.335, 1.34807,
1.36098, 1.37372, 1.38629, 1.39872, 1.41099, 1.42311, 1.43508, 1.44692, 1.45862,
1.47018, 1.4816, 1.4929, 1.50408, 1.51513, 1.52606, 1.53687, 1.54756, 1.55814,
1.56862, 1.57898, 1.58924, 1.59939, 1.60944, 1.61939, 1.62924, 1.639, 1.64866,
1.65823, 1.66771, 1.6771, 1.6864, 1.69562, 1.70475, 1.7138, 1.72277, 1.73166,
1.74047, 1.7492, 1.75786, 1.76644, 1.77495, 1.78339, 1.79176]
```

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 3. Kemudian kita membangkitkan vektor nilai fungsi.

```
lateks:s=t^3 - t
```

```
t=-1:0.1:3; 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, 0.231, 0.528, 0.897,
1.344, 1.875, 2.496, 3.213, 4.032, 4.959, 6, 7.161, 8.448, 9.867, 11.424, 13.125,
14.976, 16.983, 19.152, 21.489, 24]
```

```
shortest (1:7)(1:7)'
```

```
1 2 3 4 5 6 7 2 4 6 8 10 12 14 3 6 9 12 15 18 21 4 8 12 16 20 24 28 5 10 15
20 25 30 35 6 12 18 24 30 36 42 7 14 21 28 35 42 49
```

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

```
(1:7).(1:7)'
```

```
140
```

Secara default, vektor baris dicetak dalam format yang ringkas.

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

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

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

```
v:=[2,3]; v.v'
```

```
13 169
```

Untuk mentranspos matriks kita menggunakan apostrof

```
v=3:6; v'
```

```
3 4 5 6
```

```
A=[1,2,3,4]; A.v'
```

```
50
```

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

```
v'.v
```

```
9 12 15 18 12 16 20 24 15 20 25 30 18 24 30 36
```

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

```
v.v'
```

```
86
```

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

```
norm(v)^3
```

```
797.531190613
```

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 menduplikasikan.

Berikut ini adalah kasus sederhana dengan operator

```
[2,3,6]^2
[4, 9, 36]
```

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

```
v:=[2,3,6]; vv'
4 6 12 6 9 18 12 18 36
```

Perhatikan bahwa produk skalar menggunakan produk matriks, bukan \*!

```
v.v'
49
```

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.

```
vv, sum(vv), cumsum(vv)
[4, 9, 36] 49 [4, 13, 49]
```

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

```
3:9, 0:2:8
[3, 4, 5, 6, 7, 8, 9] [0, 2, 4, 6, 8]
[1,2]—[3,4,5], [1,2]_5
[1, 2, 3, 4, 5] 1 2 5 5
```

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

```
A:=[4,5,6,7;8,9,10,11]; A[2,2]
9
```

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

```
v:=[1,3,5,7]; v[1], A[1]
1 [4, 5, 6, 7]
```

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

```
v[2:4], A[:,2]
[3, 5, 7] 5 9
```

Bentuk singkat untuk : adalah menghilangkan indeks sepenuhnya.

```
A[:,2:4]
5 6 7 9 10 11
```

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

A5

8

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

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

```
[4, 5, 6, 7, 8, 9, 10, 11] [4, 5, 6, 7, 8, 9, 10, 11]
```

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

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

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

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

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([1:3])
```

```
[0, 2.05045, 13.7251]
```

Sub-Matriks dan Matriks-Elemen

Untuk mengakses elemen matriks, gunakan notasi braket.

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

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

Kita dapat mengakses satu baris matriks yang lengkap.

```
A[1]
```

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

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

```
v=1:8; 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[3,]
[7, 8, 9]

```

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

```

A[[2,3]]
4 5 6 7 8 9

```

Kita bahkan dapat menyusun ulang A menggunakan vektor indeks. Tepatnya, kami tidak mengubah A disini, 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[1:2,2:3]
2 3 5 6

```

Untuk singkatan ":" menunjukkan semua indeks baris atau kolom.

```

A[:,2]
2 5 8

```

Atau, biarkan indeks pertama kosong.

```

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

```

Kita juga bisa mendapatkan baris terakhir dari A.

```

A[3]
[7, 8, 9]

```

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

```

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

```

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

```

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

```

Selain itu, beberapa jalan pintas diperbolehkan.

```

A[1:2,1:2]=-1
-1 -1 3 -1 -1 5 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[5]
Row index 5 out of bounds! Error in: A[5] ...
Menyortir dan Mengacak
Fungsi sort() mengurutkan vektor baris.
sort([2,9,5,7,3,1])
[1, 2, 3, 5, 7, 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 mengacak vektor.

```
v=shuffle(1:8)
```

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

Indeks berisi urutan yang tepat dari v.

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

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

```
s=["d","f","c","b","aa","g"]
```

```
d f c b aa g
```

```
ss,ind=sort(s); ss
```

```
aa b c d f g
```

```
jpre class="udf" endfunction i/pre
```

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

Untuk sistem linier  $Ax=b$ , dengan  $A$  adalah matriks koefisien,  $x$  adalah vektor solusi yang ingin kita cari dan  $b$  adalah vektor hasil yang diberikan.

Anda dapat menggunakan algoritma Gauss, matriks invers atau kecocokan linier.

Operator `A\` menggunakan versi algoritma Gauss.

Operator `backslash` digunakan untuk menyelesaikan sistem persamaan linier ini. Ketika menulis `A\b` perangkat lunak akan menghitung solusi yang memenuhi persamaan  $Ax=b$ .

Operator ini secara otomatis menggunakan algoritma eliminasi Gauss atau metode numerik serupa untuk menemukan solusi.

```
A=[5,6;7,8]; b=[4;3]; A\b
```

```
b
```

```
-7 6.5
```

Untuk contoh lain, kami membuat matriks 100x100 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(100,100); b=sum(A); longest totalmax(abs(inv(A).b-1))
```

```
9.015010959956271e-14
```

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

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

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

Determinan matriks ini adalah `()`

```
det(A)
```

```
-34
```

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 `amp;:=`, 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
```

```
det(A),factor(
invert(A)witha = 0
```

```
A = [1,a;b,2]; A
```

dengan

‘`det(A)`’ menghitung determinan matriks A.

‘`factor(`

‘`invert(A)`’ memberikan invers dari matriks A.

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

```
det(A - xident(2)),solve(
```

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

```
eigenvalues([a, 1; 1, a])
```

Untuk mengekstrak vektor eigen tertentu perlu pengindeksan yang cermat.

```
eigenvectors([a,1;1,a]),
```

```
[[[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=6,b=7)
```

```
1 6 7 2
```

Dalam ekspresi simbolik, gunakan dengan.

```
Awith[a = 6, b = 7]
```

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

```
A[1]
```

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

```
A[1,1]:=t+1; A
```

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

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

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

```
invert(B)()
```

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

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

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

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



1 0 0 1

Misalnya. nilai eigen dari A dapat dihitung secara numerik.

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

[16.1168, -1.11684, 0]

Atau secara simbolis. Lihat tutorial tentang Maxima untuk detailnya.

`eigenvalues(@A)`

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 "amp;:=".

$A := [1, \pi; 4, 5]$

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.

$A$

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

`mxmset(A); A`

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))`

Ketepatan angka floating point besar dapat diubah

`fpprec:=100; bfloat(pi)`

3.14159265358979323846264338327950288419716939937510582097494 45923078164062862089986280348253

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)`

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 4000 (katakanlah dalam dolar).

$M = 4000$

4000

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

$M1.03$

4120

Euler akan memahami sintaks berikut juga.

$M + M3$

4120

Tetapi lebih mudah menggunakan faktornya.

```

q=1+3
1.03 4120
Selama 10 tahun, kita cukup mengalikan faktornya dan mendapatkan nilai
akhir dengan suku bunga majemuk.
Mq10
5375.66551738
Untuk tujuan kita, kita dapat mengatur format menjadi 2 digit setelah titik
desimal.
format(12,2); Mq10
5375.67
Mari kita cetak yang dibulatkan menjadi 2 digit dalam kalimat lengkap.
"Starting from " + M + "youget" + round(Mq10, 2) + "."
Starting from 4000youget5375.67.
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
Mq(0 : 10)
Real 1 x 11 matrix
4000.00 4120.00 4243.60 4370.91 ...
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.
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 q0 sampai q10. Ini dikalikan dengan M, dan kami
mendapatkan vektor nilai.
VM=Mq(0 : 10)
Real 1 x 11 matrix
4000.00 4120.00 4243.60 4370.91 ...
Tentu saja, cara realistis untuk menghitung suku bunga ini adalah dengan
membulatkan ke sen terdekat setelah setiap tahun. Mari kita tambahkan fungsi
untuk ini.
function oneyear(M) := round(M q, 2)
Mari kita bandingkan dua hasil, dengan dan tanpa pembulatan.
longest oneyear(1234.57), longest 1234.57q
1271.61 1271.6071
Sekarang tidak ada rumus sederhana untuk tahun ke-n, dan kita harus men-
gulang selama bertahun-tahun.
Euler memberikan banyak solusi untuk ini.
Cara termudah adalah iterasi fungsi, yang mengulangi fungsi tertentu be-
berapa kali.
VMr=iterate("oneyear",4000,10)
Real 1 x 11 matrix

```

4000.00 4120.00 4243.60 4370.91 ...

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

VMr'

4000.00 4120.00 4243.60 4370.91 4502.04 4637.10 4776.21 4919.50 5067.09  
5219.10 5375.67

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

VMr[2], VMr[1:3]

4120.00 4000.00 4120.00 4243.60

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.

VMr[-2], VM[-2]

5219.10 5219.09

Perbedaannya sangat kecil.

Memecahkan Persamaan

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

function onepay (M) := Mq+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",4000,10)

Real 1 x 11 matrix

4000.00 4320.00 4649.60 4989.09 ...

Bagaimana jika kita menghapus jumlah yang sama setiap tahun?

R=-200; iterate("onepay",4000,10)

Real 1 x 11 matrix

4000.00 3920.00 3837.60 3752.73 ...

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.

VMR=iterate("onepay",4000,50)

Real 1 x 51 matrix

4000.00 3920.00 3837.60 3752.73 ...

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

min(nonzeros(VMR;0))

32.00

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

Karena vektor selalu dimulai dengan indeks 1, jawabannya adalah 31 tahun.  
 Fungsi `iterate()` memiliki satu trik lagi. Itu bisa mengambil kondisi akhir sebagai argumen. Kemudian akan mengembalikan nilai dan jumlah iterasi.

```
x,n=iterate("onipay",4000,till="xj0"); x, n
-0.21 31.00
```

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(M,R,P,n) := iterate("x(1+P/100)+R",M,n;P,R)[-1]
```

Iterasinya sama seperti di atas.

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(4000,-200,3,31)
-0.21
```

Sekarang kita bisa menyelesaikan masalah kita.

```
solve("f(4000,-200,x,50)",3)
4.43
```

Rutin memecahkan memecahkan ekspresi=0 untuk variabel  $x$ . Jawabannya adalah 3,15 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

```
solve("f(4000,x,3,20)",-200)
-268.86
```

Perhatikan bahwa Anda tidak dapat memecahkan 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 tersebut. Pertama kita mendefinisikan fungsi `onipay()` kita secara simbolis.

```
function op(M) = Mq+R; op(M)
```

Kita sekarang dapat mengulangi ini.

```
op(op(op(op(M)))) , expand(
```

Kami melihat sebuah pola. Setelah  $n$  periode yang kita miliki

Rumusnya adalah rumus untuk jumlah geometri, yang diketahui Maxima.

$\text{sum}(q^k, k, 0, n - 1);$   
 Ini agak rumit. Jumlahnya dievaluasi dengan bendera "simpsum" untuk mengurangnya menjadi hasil bagi.  
 Mari kita membuat fungsi untuk ini.  
 Rumusnya adalah rumus untuk jumlah geometri, yang diketahui Maxima.  
 $\text{function fs}(M, R, P, n) = (1 + P/100)^n M + ((1 + P/100)^n - 1) / (P/100) R; \text{fs}(M, R, P, n)$   
 Fungsi tersebut melakukan hal yang sama seperti fungsi f kita sebelumnya.  
 Tapi itu lebih efektif.  
 $\text{longest f}(4000, -200, 3, 31), \text{longest fs}(4000, -200, 3, 31)$   
 $-0.2142542009293322 - 0.2142542009369208$   
 Kita sekarang dapat menggunakannya untuk menanyakan waktu n. Kapan modal kita habis? Dugaan awal kami adalah 30 tahun.  
 fungsi untuk ini.  
 Rumusnya adalah rumus untuk jumlah geometri, yang diketahui Maxima.  
 $\text{solve}("fs(4000, -330, 3, x)", 30)$   
 $15.29$   
 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  $K_n$  (pada saat pembayaran terakhir).  
 Rumus untuk ini jelas.  
 $\text{equ} = \text{fs}(M, R, P, n) = Mn; \text{equ}$   
 Biasanya rumus ini diberikan dalam bentuk  
 $\text{equ} = (\text{equ with } P=100i); \text{equ}$   
 Kita dapat memecahkan tingkat R secara simbolis.  
 $\text{solve}(\text{equ}, R)$   
 Seperti yang Anda lihat dari rumus, fungsi ini mengembalikan kesalahan titik mengambang untuk  $i=0$ .  
 Euler tetap merencanakannya.  
 Tentu saja, kami memiliki batas berikut.  
 $\text{limit}(R(4000, 0, x, 10), x, 0)$   
 Jelas, tanpa bunga kita harus membayar kembali 10 tarif 500.  
 Persamaan juga dapat diselesaikan untuk n. Kelihatannya lebih bagus, jika kita menerapkan beberapa penyederhanaan untuk itu.  
 $\text{fn} = \text{solve}(\text{equ}, n) \text{ — ratsimp}; \text{fn}$   
**LATIHAN SOAL**  
 Nama : LASIGI YATINDRA JAGO  
 Kelas : Matematika B  
 NIM : 23030630008

Pilih minimal 5 soal dari setiap Latihan atau tipe soal (misalnya diantara soal-soal yang sudah saya blok). Jangan lupa tuliskan soalnya di teks komentar (dengan format LaTeX) dan beri penjelasan hasil output-nya. Ubah file

notebook pekerjaan Anda menjadi file PDF menggunakan salah satu metode di atas.

—  
R.2 Exercise set  
Soal No 49  
Menyederhanakan:

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

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

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

Soal No 50  
Menyederhanakan:

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

$$((125p^{12}q^{-14}r^{22})/25p^8q^6r^{-15})^{-4}$$

$$\frac{q^{32}}{625 p^{80} r^{28}}$$

Soal No 90  
calculate

$$2^6 * 2^{-3} / 2^{10} / 2^{-8}$$

$$2^6 2^{-3} / 2^{10} / 2^{-8}$$

2  
Soal No 91  
Calculate

$$\left(\frac{4(8-6)^2-4*3+2*8}{3^1+9^0}\right)$$

$$(4(8-6)^2-43+28)/(3^1+9^0)$$

5  
Soal No 92  
Calculate

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

$$((4(8-6)^2+4)3-28)/(3^1+9^0)$$

11  
R.3 Exercise Set

Perform the indicated operations.

no 27

$$(x + 3)^2$$

*showev('expand((x + 3)^2))*

no 29

$$(y - 5)^2$$

*showev('expand((y - 5)^2))*

no 33

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

*showev('expand((2x + 3y)^2))*

no 39

$$(3y + 4)(3y - 4)$$

*showev('expand((3y + 4)(3y - 4)))*

no 42

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

*showev('expand((3x + 5y)(3x - 5y)))*

R.4 Exercise Set Faktor Trinomial

Nomor 24

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

*solve(y^2 + 12y + 27)*

$$[y = -9, y = -3]$$

Nomor 23

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

*solve(t^2 + 8t + 15)*

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

Factor the difference of squares

Nomor 47

$$z^2 - 81$$

*solve(z^2 - 81)*

$$[z = -9, z = 9]$$

Nomor 48

$$m^2 - 4$$

$$\text{solve}(m^2 - 4)$$

$$[m = -2, m = 2]$$

Nomor 49

$$16x^2 - 9$$

$$\text{solve}(16x^2 - 9)$$

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

R.5 Exercise Set

soal no 36

tentukan nilai y

$$y^2 - 4y - 45 = 0$$

$$\text{solve}(y^2 - 4y - 45, y)$$

$$[y = 9, y = -5]$$

soal no 38

tentukan nilai y

$$t^2 + 6t = 0$$

$$\text{solve}(t^2 + 6t, t)$$

$$[t = -6, t = 0]$$

soal no 41

tentukan nilai x

$$x^2 + 100 = 20x$$

$$\text{solve}(x^2 + 100 = 20x, x)$$

$$[x = 10]$$

soal no 42

tentukan nilai y

$$y^2 + 25 = 10y$$

$$\text{solve}(y^2 + 25 = 10y, y)$$



$$[y = 5]$$

soal no 45  
tentukan nilai y

$$3y^2 + 8y + 4 = 0$$

$$\text{solve}(3y^2 + 8y + 4, y)$$

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

soal no 47  
tentukan nilai z

$$12z^2 + z = 6$$

$$\text{solve}(12z^2 + z = 6, z)$$

$$\left[ z = -\frac{3}{4}, z = \frac{2}{3} \right]$$

soal no 60  
tentukan nilai x

$$5x^2 - 75 = 0$$

$$\text{solve}(5x^2 - 75 = 0, x)$$

$$\left[ x = -\sqrt{15}, x = \sqrt{15} \right]$$

R.6 Exercise Set  
Nomor 9  
Menyederhanakan

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

$$((x^2)/(x^2 - 4x + 4)), \text{factor}(\$$

$$\frac{x^2}{(x - 2)^2}$$

!images/ALJABAR<sub>L</sub>asigi  
Nomor 11  
Menyederhanakan

$$\frac{x^3 - 6x^2 + 9x}{x^3 - 3x^2}$$

$$((x^3 - 6x^2 + 9x)/(x^3 - 3x^2)), \text{factor}(\$$

$$\frac{x-3}{x}$$

!images/ALJABAR<sub>L</sub>asigi  
 Nomor 14  
 Menyederhanakan

$$\frac{2x^2 - 20x + 50}{10x^2 - 30x - 100}$$

$$((2x^2 - 20x + 50)/(10x^2 - 30x - 100)), \text{factor}(\$$

$$\frac{x-5}{5(x+2)}$$

!images/ALJABAR<sub>L</sub>asigi  
 Nomor 15  
 Menyederhanakan

$$\frac{4-x}{x^2+4x-32}$$

$$((4-x)/(x^2+4x-32)), \text{factor}(\$$

$$-\frac{1}{x+8}$$

!images/ALJABAR<sub>L</sub>asigi  
 Nomor 16  
 Menyederhanakan

$$\frac{6-x}{x^2-36}$$

$$((6-x)/(x^2-36)), \text{factor}(\$$

$$-\frac{1}{x+6}$$

!images/ALJABAR<sub>L</sub>asigi  
 Nomor 23

$$(((m^2 - n^2)/(r + s))/((m - n)/r + s)), \text{factor}(\$$

$$\frac{(m-n)(n+m)r}{(s+r)(rs-n+m)}$$

!images/ALJABAR<sub>L</sub>asigi  
 Nomor 25

$$(((3x+12)/(2x-8))/((x+4)^2/(x-4)^2)), \text{factor}(\$$

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

![images/ALJABAR\_Lasigi

Nomor 28

$$(((c^3 + 8)/(c^2 - 4))/((c^2 - 2c + 4)/(c^2 - 4c + 4))), \text{factor}(\$$

$$c - 2$$

![images/ALJABAR\_Lasigi

REVIEW Multiply

Nomor 73

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

$$\text{function } P(a, b, n, x) = (a^n - b^n)^x; P(a, b, n, x)$$

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

$$P(a, b, n, 3), \text{expand}(\$$

$$-b^{3n} + 3a^n b^{2n} - 3a^{2n} b^n + a^{3n}$$

![images/ALJABAR\_Lasigi

Nomor 71

$$\text{function } P(a, n) = (t^a + t^(-a))^n; P(a, n)$$

$$\left(t^a + \frac{1}{t^a}\right)^n$$

$$P(a, 2), \text{expand}(\$$

$$t^{2a} + \frac{1}{t^{2a}} + 2$$

![images/ALJABAR\_Lasigi

Soal 39

$$\text{solve}(9x^2 - 30x + 25, x)$$

$$\left[x = \frac{5}{3}\right]$$

soal nomor 41

$$\text{solve}(18x^2 - 3x + 6, x)$$

$$\left[x = \frac{1 - \sqrt{47}i}{12}, x = \frac{\sqrt{47}i + 1}{12}\right]$$

soal nomor 48

$$\text{solve}((8 - 3x) = (-7 + 2x), x)$$

$$[x = 3]$$

soal nomor 70

$$'((x^n + 10)(x^n - 4)) = \text{expand}(((x^n + 10)(x^n - 4)))$$

$$(x^n - 4) (x^n + 10) = x^{2n} + 6x^n - 40$$

soal nomor 71

$$'((t^a + t^b - a)^2) = \text{expand}(((t^a + t^b - a)^2))$$

$$\left(t^a + \frac{1}{t^a}\right)^2 = t^{2a} + \frac{1}{t^{2a}} + 2$$

soal nomor 72

$$'((y^b - z^c)(y^b + z^c)) = \text{expand}((y^b - z^c)(y^b + z^c))$$

$$(y^b - z^c) (z^c + y^b) = y^{2b} - z^{2c}$$

soal nomor 73

$$'((a^n - b^n)^3) = \text{expand}((a^n - b^n)^3)$$

$$(a^n - b^n)^3 = -b^{3n} + 3a^n b^{2n} - 3a^{2n} b^n + a^{3n}$$

soal chapter R test nomor 32

$$'(((x^2 + x - 6)/(x^2 + 8x + 15))((x^2 - 25)/(x^2 - 4x + 4))) = \text{simplify}(((x^2 + x - 6)/(x^2 + 8x + 15))((x^2 - 25)/(x^2 - 4x + 4)))$$

$$\frac{(x^2 - 25) (x^2 + x - 6)}{(x^2 - 4x + 4) (x^2 + 8x + 15)} = \text{simplify} \left( \frac{(x^2 - 25) (x^2 + x - 6)}{(x^2 - 4x + 4) (x^2 + 8x + 15)} \right)$$

$$\text{solve}(((x^2 + x - 6)/(x^2 + 8x + 15))((x^2 - 25)/(x^2 - 4x + 4)), x)$$

$$[x = 5]$$

soal nomor 33

$$'(((x)/(x^2 - 1)) - ((3)/(x^2 + 4x - 5))) = \text{simplify}(((x)/(x^2 - 1)) - ((3)/(x^2 + 4x - 5)))$$

$$\frac{x}{x^2 - 1} - \frac{3}{x^2 + 4x - 5} = \text{simplify} \left( \frac{x}{x^2 - 1} - \frac{3}{x^2 + 4x - 5} \right)$$

$$\text{solve}(((x)/(x^2 - 1)) - ((3)/(x^2 + 4x - 5)))$$

$$[x = -3]$$

2.3 Exercise Set

Cari

$$(f \circ g)(x) \text{ dan } (g \circ f)(x)$$

dan domain nya !

Nomor 17.

$$f(x) = x + 3, g(x) = x - 3$$

$$(f \circ g)(x) =$$

$$gx := x - 3; fx := gx + 3; fx$$

$$x$$

dengan domainnya

$$D_{f \circ g} = \{x \in R\}$$

$$(g \circ f)(x) =$$

$$fx := x + 3; gx := fx - 3; gx$$

$$x$$

dengan domainnya

Nomor 23.

$$gx := 1/x; fx := 4/(1-5gx); fx$$

$$\frac{4}{1 - \frac{5}{x}}$$

dengan domain

$$D_{f \circ g} = \{x \in R | x \neq 0 \cup x \neq 5\}$$

$$(g \circ f)(x) =$$

$$fx := 4/(1 - 5x); gx := 1/fx; gx$$

$$\frac{1 - 5x}{4}$$

dengan domainnya

$$D_{g \circ f} = \{x \in R\}$$

Diberikan fungsi

$$f(x) = 3x + 1, g(x) = x^2 - 2x - 6, h(x) = x^3$$

cari

Nomor 6.

$$(f \circ g)(1/3)$$

$$x := 1/3; gx := x^2 - 2x - 6; fx := 3gx + 1; fx$$

$$-\frac{56}{3}$$

Nomor 4.

$$(g \circ h) (1/2)$$

$$x := 1/2; hx := x^3; gx := hx^2 - 2hx - 6; gx$$

$$-\frac{399}{64}$$

Nomor 9.

$$(g \circ g) (-2)$$

$$x := -2; gx := x^2 - 2x - 6; gx := gx^2 - 2gx - 6; gx$$

$$-6$$

### 3.1 Exercise Set

Use the quadratic formula to find exact solutions

Nomor 37

$$x^2 - 2 = 15$$

$$\text{solve}(x^2 - 2 = 15, x)$$

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

Nomor 39

$$5m^2 + 3m = 2$$

$$\text{solve}(5m^2 + 3m = 2, m)$$

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

Nomor 40

$$2y^2 - 3y - 2 = 0$$

$$\text{solve}(2y^2 - 3y - 2 = 0, y)$$

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

Solve.

Nomor 83

$$y^4 + 4y^2 - 5 = 0$$

$$\text{solve}(y^4 + 4y^2 - 5 = 0, y)$$

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

Nomor 84

$$y^4 - 15y^2 - 16 = 0$$

$$\text{solve}(y^4 - 15y^2 - 16 = 0, y)$$

$$[y = -i, y = i, y = -4, y = 4]$$

3.4 Exercise Set

Solve

Nomor 1

$$\text{solve}(1/4 + 1/5 = 1/t, t)$$

$$\left[ t = \frac{20}{9} \right]$$

Nomor 2

$$\text{solve}(1/3 - 5/6 = 1/x, x)$$

$$[x = -2]$$

Nomor 6

$$\text{solve}(1/t + 1/2t + 1/3t = 5, t)$$

$$\left[ t = \frac{15 - \sqrt{195}}{5}, t = \frac{\sqrt{195} + 15}{5} \right]$$

Nomor 60

$$\text{solve}(\text{sqrt}(2x + 5) = x - 5, x)$$

$$[x = \sqrt{2x + 5} + 5]$$

Nomor 10

$$\text{solve}((1/5x + 20) - (1/x^2 - 16) = (3/x - 4), x)$$

$$\left[ x = \left( -\frac{\sqrt{3}i}{2} - \frac{1}{2} \right) \left( \frac{5\sqrt{6771313}i}{2 \cdot 3^{\frac{3}{2}}} - \frac{16026865}{54} \right)^{\frac{1}{3}} + \frac{40045 \left( \frac{\sqrt{3}i}{2} - \frac{1}{2} \right)}{9 \left( \frac{5\sqrt{6771313}i}{2 \cdot 3^{\frac{3}{2}}} - \frac{16026865}{54} \right)^{\frac{1}{3}}} - \frac{200}{3}, x = \left( \frac{\sqrt{3}i}{2} - \frac{1}{2} \right) \left( \frac{5\sqrt{6771313}i}{2 \cdot 3^{\frac{3}{2}}} - \frac{16026865}{54} \right)^{\frac{1}{3}} + \frac{40045 \left( \frac{\sqrt{3}i}{2} - \frac{1}{2} \right)}{9 \left( \frac{5\sqrt{6771313}i}{2 \cdot 3^{\frac{3}{2}}} - \frac{16026865}{54} \right)^{\frac{1}{3}}} - \frac{200}{3} \right]$$

Nomor 18

$$\text{solve}(3y + 5/y^2 + 5y + y + 4/y + 5 = y + 1/y, y)$$

$$\left[ y = -\frac{47 \left( \frac{\sqrt{3}i}{2} - \frac{1}{2} \right)}{576 \left( \frac{\sqrt{35539}}{128 \cdot 3^{\frac{3}{2}}} - \frac{3905}{13824} \right)^{\frac{1}{3}}} + \left( \frac{\sqrt{35539}}{128 \cdot 3^{\frac{3}{2}}} - \frac{3905}{13824} \right)^{\frac{1}{3}} \left( -\frac{\sqrt{3}i}{2} - \frac{1}{2} \right) - \frac{5}{24}, y = \left( \frac{\sqrt{35539}}{128 \cdot 3^{\frac{3}{2}}} - \frac{3905}{13824} \right)^{\frac{1}{3}} \left( \frac{\sqrt{3}i}{2} - \frac{1}{2} \right) - \frac{5}{24} \right]$$

### 3.5 Exercise Set

load(fourier\_elim)

C:/Program Files/Euler x64/maxima/share/maxima/5.35.1/share/fourier\_elim/fourier\_elim.lisp

Nomor 44

*fourier\_elim*([4x] 20, [x])//4x 20

$$[[4 (x - 5)] > 0]$$

Nomor 45

*fourier\_elim*([x + 8] < 9, [x])//x + 8 < 9

$$[[1 - x] > 0]$$

Nomor 47

*fourier\_elim*([x + 8] = 9, [x])//x + 8 = 9

$$[[x - 1] = 0] \vee [[x - 1] > 0]$$

Nomor 52

*fourier\_elim*([3x + 4] < 13, [x])//3x + 4 < 13

$$[[-3 (x - 3)] > 0]$$

Nomor 62

*fourier\_elim*([3x + 5] < 0, [x])//3x + 5 < 0

$$[[-3 x - 5] > 0]$$

### Chapter 3 Test

Nomor 8

*solve*(3/3x + 4 + 2/x - 1 = 2, x)

$$\left[ x = \frac{-\sqrt{7}i - 1}{2}, x = \frac{\sqrt{7}i - 1}{2} \right]$$

Nomor 11

*fourier\_elim*([x + 4] = 7, [x])//x + 4 = 7

$$[[x - 3] = 0]$$

Nomor 12

*fourier\_elim*([4y - 3] = 5, [x])//4y - 3 = 5

$$[[4 (y - 2)] = 0]$$

Solve

Nomor 13

*fourier\_elim*([x + 3] <= 4, [x])//x + 3 <= 4

$$[[x - 1] = 0] \vee [[1 - x] > 0]$$



Nomor 15

$$\text{fourier\_elim}([x + 5] \ 2, [x]) // x + 5 \ 2$$

$$[[x + 3] > 0]$$

Nomor 19

$$\text{solve}(x^2 + 4x = 1, x)$$

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

#### 4.1 Exercise Set

Use the substitution to determine whether 2,3 and -1 are zeros of

Nomor 23

$$\text{function } P(x) = (x^3 - 9x^2 + 14x + 24); P(x)$$

$$x^3 - 9x^2 + 14x + 24$$

$$P(4)$$

$$0$$

$$P(5)$$

$$-6$$

$$P(-2)$$

$$-48$$

Jadi hasil substitusi yang menghasilkan persamaan mempunyai nilai nol adalah dengan mensubstitusikan angka 4

Nomor 24

$$\text{function } P(x) = (2x^3 - 3x^2 + x + 6); P(x)$$

$$2x^3 - 3x^2 + x + 6$$

$$P(2)$$

$$12$$

$$P(3)$$

$$36$$

$$P(-1)$$

$$0$$

Jadi hasil substitusi yang menghasilkan persamaan mempunyai nilai nol adalah dengan mensubstitusikan angka -1

Nomor 25

$$\text{function } P(x) = (x^4 - 6x^3 + 8x^2 + 6x - 9); P(x)$$

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

$$P(2)$$

$$3$$

$$P(3)$$

$$0$$

$$P(-1)$$

0

Jadi hasil substitusi yang menghasilkan persamaan mempunyai nilai nol adalah dengan mensubstitusikan angka 3 dan -1

Nomor 37

$$\text{solve}(x^4 - 4x^2 + 3, x)$$

$$\left[ x = -1, x = 1, x = -\sqrt{3}, x = \sqrt{3} \right]$$

Nomor 39

$$\text{solve}(x^3 + 3x^2 - x - 3, x)$$

$$[x = 1, x = -1, x = -3]$$

#### 4.3 Exercise Set

Nomor 1

For the function

$$f(x) = x^4 - 6x^3 + x^2 + 24x - 20$$

use long division to determine whether each of the following is a factor of  $f(x)$

a)  $x+1$

b)  $x-2$

c)  $x + 5$

$$\text{function } f(x) = (x^4 - 6x^3 + x^2 + 24x - 20); f(x)$$

$$x^4 - 6x^3 + x^2 + 24x - 20$$

$$f(x+1), \text{expand}(\$$

$$x^4 - 2x^3 - 11x^2 + 12x$$

!images/ALJABAR<sub>L</sub>asigi

$$f(x-2), \text{expand}(\$$

$$x^4 - 14x^3 + 61x^2 - 84x$$

!images/ALJABAR<sub>L</sub>asigi

$$f(x+5), \text{expand}(\$$

$$x^4 + 14x^3 + 61x^2 + 84x$$

!images/ALJABAR<sub>L</sub>asigi

Nomor 23

Use synthetic division to find the function values.

$$f(x) = x^3 - 6x^2 + 11x - 6$$

find  $f(1)$ ,  $f(-2)$ , dan  $f(3)$

$$\text{function } f(x) = (x^3 - 6x^2 + 11x - 6); f(x)$$

$$x^3 - 6x^2 + 11x - 6$$

f(1)  
0  
f(-2)  
-60  
f(3)  
0  
Nomor 24

$$f(x) = x^3 + 7x^2 - 12x - 3$$

find f(-3),f(-2), dan f(1)  
function f(x) = (x<sup>3</sup> + 7x<sup>2</sup> - 12x - 3);f(x)

$$x^3 + 7x^2 - 12x - 3$$

f(-3)  
69  
f(-2)  
41  
f(1)  
-7  
Nomor 25

$$f(x) = x^4 - 3x^3 + 2x + 8$$

find f(-1),f(4) dan f(-5)  
function f(x) = (x<sup>4</sup> - 3x<sup>3</sup> + 2x + 8);f(x)

$$x^4 - 3x^3 + 2x + 8$$

f(-1)  
10  
f(4)  
80  
f(-5)  
998  
Factor the polynomial function f(x). Then Solve the equation f(x)=0  
Nomor 39

$$f(x) = x^3 + 4x^2 + x - 6$$

fx = (x<sup>3</sup> + 4x<sup>2</sup> + x - 6 = 0);fx

$$x^3 + 4x^2 + x - 6 = 0$$

factor((fx, x<sup>3</sup> + 4x<sup>2</sup> + x - 6 = 0))

$$(x - 1) (x + 2) (x + 3) = 0$$

$$\text{solve}(x^3 + 4x^2 + x - 6 = 0, x)$$

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

Nomor 40

$$f(x) = x^3 + 5x^2 - 2x - 24$$

$$\text{fx} = (x^3 + 5x^2 - 2x - 24 = 0); \text{fx}$$

$$x^3 + 5x^2 - 2x - 24 = 0$$

$$\text{factor}((fx, x^3 + 5x^2 - 2x - 24 = 0))$$

$$(x - 2) (x + 3) (x + 4) = 0$$

$$\text{solve}(x^3 + 5x^2 - 2x - 24 = 0, x)$$

$$[x = -4, x = -3, x = 2]$$

Mid-Chapter Mixed Review

Use synthetic division to find the function values

Nomor 18

$$g(x) = x^3 - 9x^2 + 4x - 10$$

find  $g(-5)$

$$\text{function } g(x) = (x^3 - 9x^2 + 4x - 10); g(x)$$

$$x^3 - 9x^2 + 4x - 10$$

$g(-5)$

-380

Nomor 19

$$f(x) = 20x^2 - 40x$$

find  $f(1/2)$

$$\text{function } f(x) = (20x^2 - 40x); f(x)$$

$$20x^2 - 40x$$

$f(1/2)$

-15

Using synthetic division, determine whether the numbers are zeros of the polynomial function.

Nomor 22

-1, 5;

$$f(x) = x^6 - 35x^4 + 259x^2 - 225$$

function f(x) = (x<sup>6</sup> - 35x<sup>4</sup> + 259x<sup>2</sup> - 225);f(x)

$$x^6 - 35x^4 + 259x^2 - 225$$

f(-1.5)

191.953125

Factor the polynomial function f(x). Then solve the equation f(x) = 0.

Nomor 23

$$h(x) = x^3 - 2x^2 - 55x + 56$$

hx = (x<sup>3</sup> - 2x<sup>2</sup> - 55x + 56 = 0);hx

$$x^3 - 2x^2 - 55x + 56 = 0$$

factor((hx, x<sup>3</sup> - 2x<sup>2</sup> - 55x + 56 = 0))

$$(x - 8) (x - 1) (x + 7) = 0$$

solve(x<sup>3</sup> - 2x<sup>2</sup> - 55x + 56 = 0, x)

$$[x = -7, x = 8, x = 1]$$

Nomor 24

$$g(x) = x^4 - 2x^3 - 13x^2 + 14x + 24$$

gx = (x<sup>4</sup> - 2x<sup>3</sup> - 13x<sup>2</sup> + 14x + 24 = 0);gx

$$x^4 - 2x^3 - 13x^2 + 14x + 24 = 0$$

factor((gx, x<sup>4</sup> - 2x<sup>3</sup> - 13x<sup>2</sup> + 14x + 24 = 0))

$$(x - 4) (x - 2) (x + 1) (x + 3) = 0$$

solve(x<sup>4</sup> - 2x<sup>3</sup> - 13x<sup>2</sup> + 14x + 24 = 0, x)

$$[x = -3, x = -1, x = 2, x = 4]$$

### 3 Plot 2D

Tugas Individu Nama : Lasigi Yatindra Jago

Kelas : MATEMATIKA B 2023

NIM : 23030630008

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).

#### Basic Plots

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

Macam-macam basic plot :

```
clg; // untuk membersihkan layar
window(0,0,1024,1024); // gunakan semua window
setplot(0,1,0,1); // koordinat set plot
hold on; // untuk memulai overwrite mode
n=100; X=random(n,2); Y=random(n,2); // untuk membuat koordinat
acak
```

```
colors=rgb(random(n),random(n),random(n)); // get random colors
loop 1 to n; color(colors[]); plot(X[],Y[]); end; // plot
hold off; // mengakhiri overwrite mode
insimg; // memasukkan ke notebook notebook
```

```
![images/PLOT2D_Lasigi
```

```
reset;
```

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

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

Untuk menampilkan gambar hasil plot di layar notebook, perintah `plot2d()` dapat diakhiri dengan titik dua (`:`).

Cara lain, adalah perintah `plot2d()` diakhiri dengan titik koma (`,`), kemudian menggunakan perintah `insimg()` untuk menampilkan gambar hasil plot.

Contoh lain, kita menggambar plot sebagai sisipan di plot lain. Hal ini dilakukan dengan mendefinisikan jendela plot yang lebih kecil. Perhatikan bahwa jendela ini tidak memberikan ruang untuk label sumbu di luar jendela plot. Maka Kita harus menambahkan beberapa margin sesuai kebutuhan.

```
plot2d("x3 - x");
xw=200; yw=100; ww=300; hw=300;
ow=window();
window(xw,yw,xw+ww,yw+hw);
hold on;
barclear(xw-50,yw-10,ww+60,ww+60);
plot2d("x4 - x",grid = 6) :
![images/PLOT2D_Lasigi
hold off;
window(ow);
```

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

Plot Aspect

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

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

```
aspect(1); // rasio panjang dan lebar 2:1
plot2d(["sin(x)", "cos(x)"], 0, 2pi);
!images/PLOT2D_Lasigi
aspect(2); // rasio panjang dan lebar 2:1
plot2d(["sin(x)", "cos(x)"], 0, 2pi);
!images/PLOT2D_Lasigi
aspect(3); // rasio panjang dan lebar 2:1
plot2d(["sin(x)", "cos(x)"], 0, 2pi);
!images/PLOT2D_Lasigi
aspect();
reset;
Fungsi reset() mengembalikan default plot termasuk rasio aspek.
Plot 2D di Euler
```

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

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

```
Euler dapat membuat plot 2D
* ekspresi
* fungsi, variabel, atau kurva berparameter,
* vektor nilai x-y,
* awan titik di pesawat,
* kurva implisit dengan level atau wilayah level.
* Fungsi kompleks
```

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

Plot Ekspresi atau Variabel

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

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

Catatan: Jika Anda mengakhiri baris perintah dengan titik dua ":", plot akan dimasukkan ke dalam jendela teks.

```
plot2d("x^2") :
!images/PLOT2D_Lasigi
aspect(1.5); plot2d("x^3 - x") :
!images/PLOT2D_Lasigi
a:=5.6; plot2d("exp(-ax^2)/a"); insimg(30);
!images/PLOT2D_Lasigi
```

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

Rentang plot diatur dengan parameter yang ditetapkan sebagai berikut

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

```
plot2d("x3 - x", -1, 2) :
```

```
!images/PLOT2DLasigi
```

```
plot2d("sin(x)", -2pi, 2pi):
```

```
!images/PLOT2DLasigi
```

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

```
!images/PLOT2DLasigi
```

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

Dalam opsi, plot dapat diatur agar muncul di jendela terpisah yang dapat diubah ukurannya.

Untuk membagi jendela menjadi beberapa plot, gunakan perintah figure().

Dalam contoh, kita memplot  $x^1$  hingga  $x^4$  menjadi 4 bagian jendela. gambar(0) mengatur ulang jendela default.

```
reset;
```

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

```
!images/PLOT2DLasigi
```

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

```
figure(3,3); ... for k=1:9; figure(k); plot2d("x3 - x", -2, 1, grid = k); end; ... figure(0) :
```

```
!images/PLOT2DLasigi
```

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

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

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

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

```
!images/PLOT2DLasigi
```

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

```
!images/PLOT2DLasigi
```

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

```
plot2d("sign(x)exp(-x2)", -1, 1, < adaptive, n = 10000) :
```

```
!images/PLOT2DLasigi
```

```
plot2d("xx", r = 1.2, cx = 1, cy = 1) :
```



!images/PLOT2D<sub>Lasigi</sub>

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

```
plot2d("x3 - x", square) :
```

!images/PLOT2D<sub>Lasigi</sub>

```
plot2d("integrate("sin(x)exp(-x2)", 0, x)", 0, 2) :
```

!images/PLOT2D<sub>Lasigi</sub>

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

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

!images/PLOT2D<sub>Lasigi</sub>

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

!images/PLOT2D<sub>Lasigi</sub>

```
a:=5.6; expr = exp(-a*x2)/a;
```

```
plot2d(expr, -2, 2):
```

!images/PLOT2D<sub>Lasigi</sub>

```
plot2d(expr, r=1, thickness=2):
```

!images/PLOT2D<sub>Lasigi</sub>

```
plot2d(diff(expr, x), add, style="-", color=red):
```

!images/PLOT2D<sub>Lasigi</sub>

```
plot2d(diff(expr, x, 2), a=-2, b=2, c=-2, d=1):
```

!images/PLOT2D<sub>Lasigi</sub>

```
plot2d(diff(expr, x), a=-2, b=2, square):
```

!images/PLOT2D<sub>Lasigi</sub>

```
plot2d("x2", 0, 1, steps = 1, color = red, n = 10) :
```

!images/PLOT2D<sub>Lasigi</sub>

```
plot2d("x2", add, steps = 2, color = blue, n = 10) :
```

!images/PLOT2D<sub>Lasigi</sub>

Fungsi dalam satu Parameter

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

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

```
function f(x,a) := x2/a + a*x2 - x; //define a function
```

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

!images/PLOT2D<sub>Lasigi</sub>

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

!images/PLOT2D<sub>Lasigi</sub>

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

!images/PLOT2D<sub>Lasigi</sub>

```
plot2d("f(x,b)", b=0.1, 0, 1): // plot with 0.1
```

```

!images/PLOT2DLasigi
function f(x) := x3 - x; ... plot2d("f", r = 1) :
!images/PLOT2DLasigi
Berikut ini ringkasan fungsi yang diterima
* ekspresi atau ekspresi simbolik di x
* fungsi atau fungsi simbolik dengan nama "f"
* fungsi simbolik hanya dengan nama f
Fungsi plot2d() juga menerima fungsi simbolik. Untuk fungsi simbolik, na-
manya saja yang berfungsi.
function f(x) = diff(xx, x)
x x (log(x) + 1)
plot2d(f, 0, 2):
!images/PLOT2DLasigi
Tentu saja, untuk ekspresi atau ekspresi simbolik, nama variabel sudah
cukup untuk memplotnya.
expr = sin(x)exp(-x)
- x E sin(x)
plot2d(expr, 0, 3pi):
!images/PLOT2DLasigi
function f(x) = xx;
plot2d(f, r=1, cx=1, cy=1, color=blue, thickness=2);
plot2d(diff(f(x), x), add, color=red, style="-.-"):
!images/PLOT2DLasigi
Untuk gaya garis ada berbagai pilihan.
* gaya="...". Pilih dari "-", "--", "-.", ".", ".-", "-.-".
* Warna: Lihat di bawah untuk warna.
* ketebalan: Defaultnya adalah 1.
Warna dapat dipilih sebagai salah satu warna default, atau sebagai warna
RGB.
* 0..15: indeks warna default.
* konstanta warna: putih, hitam, merah, hijau, biru, cyan, zaitun, * abu-abu
muda, abu-abu, abu-abu tua, oranye, hijau muda, pirus, biru * muda, oranye
muda, kuning
* rgb(merah, hijau, biru): parameternya real di [0,1].
plot2d("exp(-x2)", r = 2, color = red, thickness = 3, style = " - - ") :
!images/PLOT2DLasigi
Berikut adalah tampilan warna EMT yang telah ditentukan sebelumnya.
aspect(2); columnsplot(ones(1,16), lab=0:15, grid=0, color=0:15):
!images/PLOT2DLasigi
Tapi Anda bisa menggunakan warna apa saja.
columnsplot(ones(1,16), grid=0, color=rgb(0,0,linspace(0,1,15))):
!images/PLOT2DLasigi
Menggambar Beberapa Kurva pada bidang koordinat yang sama
Memvisualisasikan data yang memiliki lebih dari satu fungsi ke dalam satu
jenis jendela/gambar yang sama, dapat dilakukan dengan beberapa cara. Salah

```

satu caranya adalah dengan menggunakan `gt;add`. `gt;add` ini digunakan untuk memanggil fungsi-fungsi tadi secara bersamaan. Kita telah menggunakan metode ini diatas, pada contoh contoh sebelumnya.

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

```
!images/PLOT2D_Lasigi
```

\* Baris perintah diatas digunakan untuk membuat grafik dari fungsi \* "`cos(x)`".

\* `aspect()` digunakan untuk mengatur rasio dari grafik yang menentukan \* proporsi antara sumbu x dan sumbu y.

\* `r=2`, artinya radius di sekitar pusat plotnya adalah 2

\* `grid=6`, digunakan untuk mengatur jumlah garis grid atau jenis grid \* yang digunakan pada grafik.

Fungsi kedua yang akan kita plot adalah fungsi `y=x`

- `style="."`, artinya kita ingin membuat grafik dengan menggunakna titik titik untuk menandai nilai-nilai pada grafik.

```
aspect(2.5); plot2d("cos(x)",r=2,grid=6); plot2d("x",style="-", add):
```

```
!images/PLOT2D_Lasigi
```

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

```
!images/PLOT2D_Lasigi
```

\* Pada perintah ini kita menggunakan `aspect(1.5)`. Fungsi yang ingin \* kita plot adalah fungsi `sin(x)` dengan nilai `x=0` sampai `x=2pi`. Fungsi \* kedua yang akan kita tambahkan ke plot `sin(x)` adalah fungsi `cos(x)` \* dengan style garisnya berupa garis putus putus berwarna biru.

Salah satu kegunaan `gt;add` adalah untuk menambahkan titik pada kurva.

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

```
!images/PLOT2D_Lasigi
```

\* Pada perintah ini kita akan memplot grafik dari fungsi `sin(x)` dengan \* interval `x` dari 0 sampai `pi`. Lalu kita ingin menambahkan suatu titik \* pada grafik sinus `x` tadi, dengan cara menambahkan perintah lain yaitu \* `plot2d(2, sin(2), gt;point, gt;add);`. 2 dalam perintah tersebut artinya \* nilai `x` yang kita pilih adalah 2. Dengan menambahkan `gt;point` pada garis \* perintah, kita sudah dapat menambahkan satu titik pada grafik sinus \* tadi.

Pada contoh di bawah ini, kita akan memplot suatu fungsi dan menambahkan suatu titik dengan label nama ("`cl`" atau center left), dan menyimpan hasilnya di notebook. Kita juga akan menambahkan label atau judul untuk plot fungsinya.

```
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):
```

```
!images/PLOT2D_Lasigi
```

\* Fungsi yang kita plot di atas adalah fungsi `cos(x)` dan `y=x`, dengan \* rentang sumbu `x` dan sumbu `y` 1.1, artinya sumbu `x` dan `y` akan diperluas \* sedikit lebih dari -1.1 hingga 1.1, posisi pusat grafik pada koordinat \* adalah 0.5 relatif terhadap rentang grafik, kurva `cos(x)` berwarna \* hitam dengan style garis solid dan kurva `y=x` berwarna biru dengan \* style titik-titik. Tampilan grid yang digunakan dalam plot ini adalah \* grid 1.

\* (x0=solve("cos(x)-x",1) merupakan perintah yang digunakan untuk \* menyelesaikan persamaan  $\cos(x)-x=0$ , dan hasilnya akan disimpan dalam \* variabel x0. Perintah plot yang kedua digunakan untuk menambahkan \* titik pada plot yang sudah dibuat sebelumnya, dengan nilai  $x=y=x0$ . \* Selain itu, plot kedua ini ditambahkan label nama "Intersection Demo".

\* (label("cos(x)=x",x0,x0,pos="C1", offset=20) merupakan perintah \* untuk menambahkan label pada titik potong anatar grafik  $\cos(x)$  dan  $x$ . \* Label nama ini diposisikan di center left, atau sebelah kiri pusat \* (titik) dan ada offset=20 dari titik potong, supaya titik tidak \* tertutup oleh label teks.

Dalam contoh berikut ini, kita akan memplot fungsi  $\text{sinc}(x)=\sin(x)/x$  dan ekspansi Taylor ke-8 dan ke-16. Untuk mencari ekspansi Taylor ini, kita akan menggunakan Maxima melalui ekspresi simbolik. Dalam perintah berikut, dilakukan pemanggilan plot2d() sebanyak 3 kali dan dilakukan dalam perintah multi baris. Perintah plot kedua dan ketiga memiliki set flag gt;add, yang membuat plot menggunakan nterval sebelumnya pada pemanggilan plot pertama.

Kita menambahkan kotak label yang menjelaskan fungsi-sungsi tersebut.  
 $\text{taylor}(\sin(x)/x, x, 0, 4)$

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

```
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]):
```

!images/PLOT2D\_Lasigi

\* Pemanggilan plot pertama digunakan untuk menggambar fungsi  $\text{sinc}(x)$  \* pada interval 0 sampai  $4\pi$ . Warna dari grafik ini adalah hijau, dengan \* ketebalan 2.

\* Pemanggilan plot kedua digunakan untuk menggambar grafik dari \* polinomial Taylor, dari derajat 0 sampai dengan derajat 8. Warna dari \* grafik ini adalah biru, dan style garis putus putus.

\* Pemanggilan plot ketiga digunakan untuk menggambar grafik dari \* polinomial Taylor, dari derajat 0 sampai dengan derajat 16. Warna dari \* grafik ini adalah biru, dan style garis dengan titik-titik putus.

\* Perintah keempat digunakan untuk menambahkan kotak label untuk \* menjelaskan grafik-grafik yang digambar. Fungsi  $\text{sinc}(x)$  diberikan \* warna hitam dengan style garis solid, fungsi Taylor sampai derajat \* ke-8 diberikan warna biru dengan style garis putus-putus, dan fungsi \* Taylor sampai derajat ke-16 diberikn warna merah dengan style garis \* garis dengan titik-titik putus.

Pada contoh berikut, kami menghasilkan Polinomial Bernstein.

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

```
plot2d("(1-x)^10",0,1) : //plot first function
```

!images/PLOT2D\_Lasigi

```
for i=1 to 10; plot2d("bin(10,i)x^i(1-x)^(10-i)", add); end;
insimg;
```

!images/PLOT2D<sub>Lasigi</sub>  
 \* Perintah (plot2d(" (1-x)<sup>10</sup>", 0, 1)) digunakan untuk menggambarkan grafik dari fungsi  $(1-x)^{10}$  pada rentang 0 hingga 1. Grafik ini menggambarkan bentuk polinomial dari fungsi tersebut.  
 \* Perintah kedua digunakan untuk mengulang perintah dari i=1 hingga i=10. Dalam setiap iterasi loop, perintah ini menggambar grafik dari suku polinomial dalam ekspansi binomial dari  $(1-x)^{10}$ . Fungsi yang diplot adalah  $\text{bin}(10, i) \cdot x^i \cdot (1-x)^{(10-i)}$  di mana  $\text{bin}(10, i)$  adalah koefisien binomial.  
 \* Perintah insimg digunakan untuk menyisipkan atau menampilkan gambar dalam hasil output.

Metode kedua menggunakan sepasang matriks nilai x dan matriks nilai y dengan ukuran yang sama.

Kita membuat sebuah matriks nilai dengan satu Polinomial Bernstein di setiap baris. Untuk ini, kita cukup menggunakan vektor kolom i. Baca kembali pengantar tentang bahasa matriks untuk mempelajari lebih lanjut.

```
x=linspace(0,1,500);
n=10; k=(0:n)'; // n is row vector, k is column vector
y=bin(n,k)xk(1-x)(n-k); // y is a matrix then
plot2d(x,y):
!images/PLOT2DLasigi
```

\* Perintah linspace digunakan untuk membuat vektor x yang berisi 500 titik yang terdistribusi secara merata (linier) dari 0 hingga 1.

\* Perintah kedua digunakan untuk membuat vektor.

\* Perintah ketiga digunakan untuk menghitung nilai-nilai untuk y berdasarkan ekspansi binomial. Hasil dari operasi pada baris perintah \* ini adalah sebuah matriks.

\* plot2d(x,y) digunakan untuk menggambar grafik 2d dari matriks y terhadap vektor x.

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):
!images/PLOT2DLasigi
```

Cara lain adalah dengan menggunakan vektor ekspresi (string). Lalu kalian dapat menggunakan larik warna, larik gaya, dan larik ketebalan dengan panjang yang sama.

```
plot2d(["sin(x)", "cos(x)"], 0, 2pi, color=2:3):
!images/PLOT2DLasigi
```

Perintah diatas digunakan untuk memplot fungsi sin(x) dan cos(x) dalam rentang 0 hingga 2pi. Pada plot in, kedua fungsi diberikan warna yang berbeda.

```
plot2d(["sin(x)", "cos(x)"], 0, 2pi): // plot vector of expressions
!images/PLOT2DLasigi
```

Kita bisa mendapatkan vektor seperti itu dari Maxima menggunakan makelist() dan mxm2str().

```
v = makelist(binomial(10,i)xi(1-x)(10-i), i, 0, 10) // makelist
10 9 8 2 7 3 [(1-x), 10 (1-x) x, 45 (1-x) x2, 120 (1-x) x3, 6 4 5 5 4 6 3
7 210 (1-x) x4, 252 (1-x) x5, 210 (1-x) x6, 120 (1-x) x7, 2 8 9 10 45 (1-x)
x8, 10 (1-x) x9, x]
```

\* fungsi makelist() digunakan untuk membuat list dari elemen-elemen \* yang dihasilkan oleh ekspresi di dalamnya berdasarkan parameter yang \* diberikan.  
 \* (binomial(10,i)) merupakan fungsi yang digunakan untuk menghitung \* koefisien binomial, yaitu angka yang muncul dalam ekspansi dari  $(a+b)^10$ .  
 \* (i,0,10) merupakan rentang iterasi untuk i dalam fungsi makelist. \* Artinya i akan berubah dari 0 hingga 10 dan untuk setiap nilai i \* dalam rentang ini, ekspresi binomial(10,i)\* $x^i(1-x)^{10-i}$  akan \* dihitung dan dimasukkan ke dalam list.

mxm2str(v) // get a vector of strings from the symbolic vector  
 (1-x)<sup>10</sup>10\*(1-x)<sup>9</sup>\*x45\*(1-x)<sup>8</sup>\*x<sup>2</sup>120\*(1-x)<sup>7</sup>\*x<sup>3</sup>210\*(1-x)<sup>6</sup>\*x<sup>4</sup>252\*(1-x)<sup>5</sup>\*x<sup>5</sup>210\*(1-x)<sup>4</sup>\*x<sup>6</sup>120\*(1-x)<sup>3</sup>\*x<sup>7</sup>45\*(1-x)<sup>2</sup>\*x<sup>8</sup>10\*(1-x)\*x<sup>9</sup>x<sup>10</sup>  
 \* mxmstr() merupakan fungsi yang digunakan untuk mengkonversi atau \* mengubah format data dari bentuk simbolik menjadi bentuk string.

plot2d(mxm2str(v),0,1): // plot functions  
 ![images/PLOT2D\_Lasigi  
 \* Perintah diatas digunakan untuk memvisualisasikan fungsi yang sudah \* dikonversi sebelumnya, yaitu fungsi mxm2str(v), dari rentang x=0 \* sampai x=1.

Alternatif lain adalah dengan menggunakan bahasa matriks di Euler.  
 Jika sebuah ekspresi menghasilkan sebuah matriks fungsi, dengan satu fungsi di setiap baris, semua fungsi ini akan diplot ke dalam satu plot.

Untuk ini, gunakan vektor parameter dalam bentuk vektor kolom. Jika sebuah larik warna ditambahkan, maka akan digunakan untuk setiap baris plot.  
 n=(1:10)'; plot2d("x^n", 0, 1, color = 1 : 10) :

![images/PLOT2D\_Lasigi  
 Ekspresi dan fungsi satu baris dapat melihat variabel global.

Jika kalian tidak dapat menggunakan variabel global, kalian perlu menggunakan fungsi dengan parameter ekstra, dan memberikan parameter ini sebagai parameter semicolon atau titik koma.

Hati-hati dalam meletakkan semua parameter yang diberikan di akhir perintah plot2d. Pada contoh di bawah ini, kita memasukkan nilai a=5 ke dalam fungsi f, yang kita plot dari -10 hingga 10.

function f(x,a) := 1/aexp(-x<sup>2</sup>/a); ...plot2d("f", -10, 10; 5, thickness = 2, title = "a = 5") :

![images/PLOT2D\_Lasigi

\* Plot diatas merupakan plot dari fungsi  $f(x,a) := 1/a \cdot \exp(-x^2/a)$ ,  $x$  merupakan variabel independen dan  $a$  merupakan variabel dependen. Interval sumbu  $x$  dari  $x = -10$  hingga  $x = 10$ , ketebalan garis kurva 2, dengan  $a = 5$ .

Atau gunakan koleksi dengan nama fungsi dan semua parameter tambahan. List atau daftar khusus ini disebut koleksi panggilan, dan itu merupakan cara yang lebih banyak digunakan untuk mengoper argumen ke fungsi yang dnegan sendirinya dioper sebagai argumen ke fungsi lain.

Pada contoh berikut, kita menggunakan perulangan untuk memplot beberapa fungsi.

plot2d("f",1,-10,10); ... for a=2:10; plot2d("f",a, add); end:  
 ![images/PLOT2D\_Lasigi

\* Plot diatas merupakan plot dari fungsi f dengan parameter a=1, dan \* dari interval x=-10 sampai x=10.

\* Perintah kedua merupakan perulangan untuk menggambar grafik fungsi f \* dengan parameter a=2:10 (2 sampai 10). Grafik baru dengan nilai a=2:10 \* ini kemudian ditambahkan ke grafik pertama saat nilai a=1.

Kita dapat mendapatkan hasil yang sama dengan grafik diatas menggunakan cara berikut, yaitu menggunakan bahasa matriks EMT. Masing-masing matriks f(x,a) adalah satu fungsi. Selain itu, kita dapat mengatur masing-masing baris dari matriks menggunakan warna yang berbeda. Klik dua kali pada fungsi getspectral() untuk penjelasan lebih lanjut.

```
x=-10:0.01:10; a=(1:10)'; plot2d(x,f(x,a),color=getspectral(a/10)):
```

```
![images/PLOT2D_Lasigi
```

\* Perintah x=-10:0.01:10 digunakan untuk membuat vektor x yang berisi \* nilai dari -10 hingga 10 dengan interval 0.01.

\* Perintah a=(1:10)' digunakan untuk mendefinisikan vektor a yang \* berisi nilai dari 10 hingga 10.

\* Perintah plot2d(x,f(x,a),color=getspectral(a/10)) digunakan untuk \* menggambar grafik dari fungsi f dengan x sebagai variabel independen \* dan a sebagai parameter. getspectral() digunakan untuk menetapkan \* warna yang berbeda pada setiap kurva.

Soal Latihan Tambahan

1. Sketsakan grafik fungsi berikut di interval 1:10

$$g(x) = \sqrt{(x+3)} + 1$$

```
function g(x) := sqrt((x+3)+a^(a-1)); ... for a = 1 : 5; plot2d("g", 1, 10, title = "Grafik g(x)"); end :
```

```
![images/PLOT2D_Lasigi
```

2. Carilah grafik dari fungsi berikut pada interval [-pi,2pi]

$$y = \sin\left(t - \frac{t}{4}\right)$$

```
function y(t) := sin(t-(t/4)); ... plot2d("y",-pi,2pi,color=blue,title="Grafik y(t)");
```

```
![images/PLOT2D_Lasigi
```

Kuis

1. Selidiki dimanakah fungsi f(x) berikut naik dan turun

$$f(x) = \left(\frac{1}{3}\right)x^3 - x^2_3x + 4$$

Label Teks

Dekorasi sederhana pun bisa

\* judul dengan title= "..."

\* label x dan y dengan xl="...", yl="..."

\* label teks lain dengan label("...",x,y)

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

```
plot2d("x3 - x", -1, 2, title = "y = x3 - x", yl = "y", xl = "x") :
!images/PLOT2D_Lasigi
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");
!images/PLOT2D_Lasigi
```

Ada juga fungsi labelbox(), yang dapat menampilkan fungsi dan teks. Dibutuhkan vektor string dan warna, satu item untuk setiap fungsi.

```
function f(x) = x2exp(-x2); ... 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) :
!images/PLOT2D_Lasigi
```

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

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

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

```
n=10; plot2d(0:n, bin(n, 0:n), addpoints); ... labelbox("Binomials", styles = "", points, x=0.1, y=0.1,
... tcolor=black, left):
!images/PLOT2D_Lasigi
```

Gaya plot ini juga tersedia di statplot(). Seperti di plot2d() warna dapat diatur untuk setiap baris plot. Masih banyak lagi plot khusus untuk keperluan statistik (lihat tutorial tentang statistik).

```
statplot(1:10, random(3, 10), color = [red, blue, green]):
!images/PLOT2D_Lasigi
Fitur serupa adalah fungsi textbox().
```

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

```
function f(x) = exp(-x)sin(2pix); ... plot2d("f(x)", 0, 2pi); ... textbox(latex("
textExample of a damped oscillation
f(x)=e-xsin(2
pix)"), w = 0.85) :
!images/PLOT2D_Lasigi
```

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

```
plot2d("x3 - x", title = u"xrarr; xsup3; -x") :
!images/PLOT2D_Lasigi
```

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

```
plot2d("sinc(x)", 0, 2pi, xl = "x", yl = u"x rarr; sinc(x)", vertical):
!images/PLOT2D_Lasigi
LaTeX
```



Anda juga dapat memplot rumus LaTeX jika Anda telah menginstal sistem LaTeX. Saya merekomendasikan MiKTeX. Jalur ke biner "lateks" dan "dvi2ps" harus berada di jalur sistem, atau Anda harus mengatur LaTeX di menu opsi.

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

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

```
plot2d("exp(-x)sin(x)/x",a=0,b=2pi,c=0,d=1,grid=6,color=blue, ... title=latex("
textFunction
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) :
```

!images/PLOT2D\_Lasigi

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

Cara termudah adalah membuat plot kosong dengan bingkai menggunakan `grid=4`, lalu menambahkan grid dengan `ygrid()` dan `xgrid()`. Pada contoh berikut, kami menggunakan tiga string LaTeX untuk label pada sumbu x dengan `xtick()`.

```
plot2d("sinc(x)",0,2pi,grid=4,xticks); ... ygrid(-2:0.5:2,grid=6); ... xgrid([0:2] *
pi,xticks,grid=6); ... xtick([0,pi,2pi],[ "0", "
pi", "2
pi"], latex):
```

!images/PLOT2D\_Lasigi

Tentu saja fungsinya juga bisa digunakan.

function map f(x) ...

if x<0 then return x^4 else return x^2 end if end function < /pre > Parameter" map" membantumenggunakan f

plot, itu tidak perlu. Tapi untuk menunjukkan vektorisasi itu

berguna, kita menambahkan beberapa poin penting ke plot di  $x=-1$ ,  $x=0$

dan  $x=1$ .

Pada plot berikut, kami juga memasukkan beberapa kode LaTeX. Kami menggunakannya untuk

dua label dan kotak teks. Tentu saja, Anda hanya bisa menggunakannya

LaTeX jika Anda telah menginstal LaTeX dengan benar.

```
plot2d("f",-1,1,xl="x",yl="f(x)",grid=6); ... plot2d([-1,0,1],f([-1,0,1]), points, add);
... label(latex("x^3"), 0.72, f(0.72)); ... label(latex("x^2"), -0.52, f(-0.52), pos =
"ll"); ... textbox(... latex("f(x) =
begin cases x^3 x 0
```

```

 $x^2x$ 
le0
endcases"), ...  $x = 0.7, y = 0.2$ ) :
!images/PLOT2D_Lasigi

```

#### Interaksi Pengguna

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

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

Tombol spasi akan mengatur ulang plot ke jendela plot aslinya.

Saat memplot data, flag `gt`; user hanya akan menunggu penekanan tombol.  
`plot2d("x3 - ax", a = 1, user, title = "Press any key!") :`

Interrupt with Escape key! Try "trace errors" to inspect local variables after errors. `plot2d: k1=mouse("Press return, cursor key, +/ -, space, click with m... plot2d("exp(x)sin(x)", user=true, ... title="+/- or cursor keys (return to exit)") :`

Interrupt with Escape key! Try "trace errors" to inspect local variables after errors. `plot2d: k1=mouse("Press return, cursor key, +/ -, space, click with m...`

Berikut ini menunjukkan cara interaksi pengguna tingkat lanjut (lihat tutorial tentang pemrograman untuk detailnya).

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

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

```

function plotf(xp,yp,select) ...
d=interp(xp,yp); plot2d("interpval(xp,d,x)",d,xp,r=2); plot2d(xp,yp,i,points,i,add);
if select<0 then plot2d(xp[select],yp[select],color=red,i,points,i,add); endif; ti-
tle("Drag one point, or press space or return!"); endfunction i/pre i
Perhatikan parameter titik koma di plot2d (d dan xp), yang diteruskan ke evaluasi fungsi interp(). Tanpa ini, kita harus menulis fungsi plotinterp() terlebih dahulu, mengakses nilainya secara global.

```

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

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

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

Pertama kita membutuhkan fungsi `plot`.

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

Kemudian kita memerlukan nama untuk parameter, nilai awal dan matriks rentang nx2, opsional garis judul.

Ada penggeser interaktif, yang dapat menetapkan nilai oleh pengguna. Fungsi `dragvalues()` menyediakan ini.

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

```
Error in return result. plotf: useglobal; return plot2d("exp(a*x)*cos(2pi*b*x)",0,2pi;a,b):
... Try "trace errors" to inspect local variables after errors. dragvalues: f(vv,args());
```

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

```
function plotf(n) ...
plot2d("cos(x)",0,2pi,i,square,grid=6); plot2d("taylor(cos(x),x,0,@n)",color=blue,i,add);
textbox("Taylor polynomial of degree "+n,0.1,0.02,style="t",i,left); endfunction
i/pre i plotf(1):
```

Sekarang kita izinkan derajat `n` bervariasi dari 0 hingga 20 dalam 20 perhentian. Hasil `dragvalues()` digunakan untuk memplot sketsa dengan `n` ini, dan untuk memasukkan plot ke dalam buku catatan.

```
nd=dragvalues("plotf","degree",2,[0,20],20,y=0.8, ... heading="Drag the
value:"); ... plotf(nd):
```

Berikut ini adalah demonstrasi sederhana dari fungsinya. Pengguna dapat menggambar jendela plot, meninggalkan jejak titik.

```
function dragtest ...
plot2d(none,r=1,title="Drag with the mouse, or press any key!"); start=0;
repeat flag,m,time=mousedrag(); if flag==0 then return; endif; if flag==2 then
hold on; mark(m[1],m[2]); hold off; endif; end endfunction i/pre i dragtest //
lihat hasilnya dan cobalah lakukan!
```

### Gaya Plot 2D

Secara default, EMT menghitung penanda kecil sumbu otomatis dan menambahkan label ke setiap penanda. Ini dapat diubah dengan parameter tampilan. Gaya default sumbu dan label dapat diubah. Selain itu, label dan judul dapat ditambahkan secara manual. Untuk menyetel ulang ke gaya default, gunakan `reset()`.

```
aspect();
figure(3,4); ... figure(1); plot2d("x3-x",grid=0);...//tidakadatampilan,bingkaidansumbu
figure(2); plot2d("x3-x",grid=1);...//terdapatsumbu-x-y
figure(3); plot2d("x3-x",grid=2);...//terdapatpenandakecilotomatis
figure(4); plot2d("x3-x",grid=3);...//sumbu-x-ydenganlabeldidalamnya
figure(5); plot2d("x3-x",grid=4);...//tidakadapenandahanyalabel
figure(6); plot2d("x3-x",grid=5);...//defaulttitidakadamargin
figure(7); plot2d("x3-x",grid=6);...//hanyasumbudanpenandakecil
figure(8); plot2d("x3-x",grid=7);...//hanyasumbudanpenandakecilperincipadasumbutertentu
figure(9); plot2d("x3-x",grid=8);...//defaultdenganpenanda-penandakecil didalamny
figure(10); plot2d("x3-x",grid=9);...//defaultdenganpenanda-penandakecil didalamny
figure(11); plot2d("x3-x",grid=10);...//tidakadapenandakecil,hanyasumbu
figure(0):
```

!images/PLOT2D<sub>Lasigi</sub>

Parameter `lt;frame` mematikan frame, dan `framecolor=blue` mengatur frame menjadi warna biru.

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

```
aspect(1.5);
plot2d("x3 - x", grid = 0); //plot
frame; xgrid([-1,0,1]); ygrid(0): // add frame and grid
!images/PLOT2DLasigi
```

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=ex")); //titleabovetheplot
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
!images/PLOT2DLasigi
```

Sumbu dapat digambar secara terpisah dengan `xaxis()` dan `yaxis()`.

```
plot2d("x3 - x", < grid, < frame);
xaxis(0,xx=-2:1,style="- "); yaxis(0,yy=-5:5,style="- ");
!images/PLOT2DLasigi
```

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

```
function f(x) = x3 - x
3 x - x
plot2d(f,-1,1,square);
x0=fmin(f,0,1); // compute point of minimum
label("Rel. Min.",x0,f(x0),pos="lc"): // add a label there
!images/PLOT2DLasigi
```

Ada juga kotak teks.

```
plot2d(f,-1,1,-2,2); // function
plot2d(diff(f(x),x), add,style="-",color=red); // derivative
labelbox(["f","f'"],["-", "-"],[black,red]): // label box
!images/PLOT2DLasigi
plot2d(["exp(x)","1+x"],color=[black,blue],style=["-", "-.-"]):
!images/PLOT2DLasigi
```

```
gridstyle("-",color=gray,textcolor=gray,framecolor=gray); ... plot2d("x3 -
x", grid = 1); ... setttitle("y = x3 - x", color = black); ... label("x", 2, 0, pos =
"bc", color = gray); ... label("y", 0, 6, pos = "cl", color = gray); ... reset() :
```

!images/PLOT2D<sub>Lasigi</sub>

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

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

```
fullwindow; ... gridstyle(color=darkgray,textcolor=darkgray); ... plot2d(["2x", "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(["2x", "1", "2-x"], colors = 4 : 6, x = 0.8, y = 0.2); ... reset :
```

!images/PLOT2D<sub>Lasigi</sub>

Berikut adalah contoh lain, di mana string Unicode digunakan dan sumbunya berada di luar area plot.

```
aspect(1.5);
plot2d(["sin(x)", "cos(x)"], 0, 2pi, color=[red, green], igrid, iframe); ... xaxis(-1.1, (0:2)pi, xt=["0", "u"pi, "u"2pi], style="-", ticks, zero); ... xgrid((0:0.5:2) × pi, i ticks); ... yaxis(-0.1pi, -1:0.2:1, style="-", zero, grid); ... labelbox(["sin", "cos"], colors=[red, green], x=0.5, y=... xlabel(u"phi"); ylabel(u"f(phi)");
```

!images/PLOT2D<sub>Lasigi</sub>

Memplot Data 2D

Jika  $x$  dan  $y$  adalah vektor data, maka data dalam vektor  $x$  dan  $y$  ini akan digunakan sebagai koordinat  $x$  dan  $y$  dari sebuah kurva. Dalam hal ini,  $a$ ,  $b$ ,  $c$ , dan  $d$ , atau radius  $r$  dapat ditentukan, jika nilai-nilai tersebut tidak ditentukan, plot window akan menyesuaikan secara otomatis dengan data. Sebagai alternatif, perintah `gt;square` dapat digunakan untuk mempertahankan rasio aspek persegi.

Memplot sebuah ekspresi hanyalah singkatan untuk plot data. Untuk data plot, kalian membutuhkan satu atau beberapa baris nilai  $x$ , dan satu atau beberapa baris nilai  $y$ . Dari rentang dan nilai  $x$ , fungsi `plot2d` akan menghitung data untuk diplot secara default dengan evaluasi adaptif dari fungsi tersebut. Untuk menambahkan titik pada gambar grafik, gunakan perintah `"gt;points"`, untuk garis dan titik gunakan perintah `"gt;addpoints"`.

Namun, kalian juga 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(-x2)x; plot2d(x, y) :
```

!images/PLOT2D<sub>Lasigi</sub>

- \* `x=-10:0.1:10`, merupakan perintah untuk membuat sebuah vektor dengan nama  $x$ , yang berisi deretan nilai dari -10 hingga 10 dengan beda \* setiap nilainya adalah 0.1.

- \* `y=exp(-x2) * x`, merupakan perintah untuk membuat vektor dengan nama  $y$ , yang berisi deretan nilai berdasarkan  $x$ .

- \* `plot2d(x, y)`, merupakan perintah untuk membuat grafik 2d dari data  $x$  \* dan  $y$ .

Data juga bisa diplot sebagai titik. Gunakan perintah `points=true` untuk membuatnya. Plot ini bekerja seperti poligon, namun hanya menggambar sudut-sudutnya saja.

```
* style="...": Select from "[]", "lt;gt;", "o", ".", "..", "+", "*", "* []", "lt;gt;", "o", ".", "..", "+", "*".
```

Untuk memplot kumpulan titik, gunakan perintah `gt;points`. Jika warnanya merupakan vektor warna, setiap titik akan mendapatkan warna yang berbeda. Untuk matriks koordinat dan vektor kolom, warna tersebut diaplikasikan pada baris setiap matriks. Parameter `gt;addpoints` menambahkan titik-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
!images/PLOT2DLasigi
```

Perintah diatas digunakan untuk membuat plot 2d dari `xdata` dan `ydata`, dengan rentang sumbu `x` dari 0.5 - 4.5 dan rentang sumbu `y` dari 2.5 - 3.5. Kumpulan titik-titik data tersebut akan ditampilkan dengan style titik.

```
plot2d(xdata,ydata, points, add,style="o"): // add points
!images/PLOT2DLasigi
p=polyfit(xdata,ydata,1); // get regression line
```

Perintah diatas digunakan untuk menghitung koefisien polinomial orde 1, dimana hasilnya adalah nilai `p` yang berisi dua koefisien dari garis regresi. Dua koefisien ini merupakan kemiringan dan intercept dari garis regresi. Garis regresi adalah garis lurus yang menggambarkan hubungan antara dua variabel, yaitu variabel dependen (`y`) dan variabel independen (`x`). Garis regresi digunakan untuk menggambarkan perilaku sekumpulan data.

```
plot2d("polyval(p,x)", add,color=red): // add plot of line
!images/PLOT2DLasigi
```

Perintah ini digunakan untuk menambahkan garis regresi yang telah dihitung sebelumnya ke dalam grafik yang sudah ada.

- perintah `"polyval(p,x)"` menghitung nilai prediksi berdasarkan koefisien polinomial `p` dan data `x`. Fungsi ini menggunakan nilai koefisien dari `p` untuk menghitung nilai `y` yang sesuai dengan regresi linier.

Menggambar Daerah Yang Dibatasi Kurva

Plot data sebenarnya berbentuk poligon. Kita juga dapat memplot kurva atau kurva terisi.

\* `fillcolor=benar` mengisi plot.

\* `style="..."`: Pilih dari `"", "/", ";", "`".

\* `fillcolor` : Lihat di atas untuk mengetahui warna yang tersedia.

Warna isian ditentukan oleh argumen `"fillcolor"`, dan pada `lt;outline` optional, mencegah menggambar batas untuk semua gaya kecuali gaya default.

```
t=linspace(0,2pi,1000); // parameter for curve
x=sin(t)exp(t/pi); y=cos(t)exp(t/pi); // x(t) and y(t)
figure(1,2); aspect(16/9)
figure(1); plot2d(x,y,r=10); // plot curve
figure(2); plot2d(x,y,r=10, filled,style="/", fillcolor=red); // fill curve
figure(0):
!images/PLOT2DLasigi
```

Dalam contoh berikut kita memplot elips terisi dan dua segi enam terisi menggunakan kurva tertutup dengan 6 titik dengan gaya isian berbeda.

```
x=linspace(0,2pi,1000); plot2d(sin(x),cos(x)0.5,r=1, filled,style="/"):
!images/PLOT2DLasigi
```

```

t=linspace(0,2pi,6); ... plot2d(cos(t),sin(t), filled,style="/" ,fillcolor=red,r=1.2):
!images/PLOT2DLasigi
t=linspace(0,2pi,6); plot2d(cos(t),sin(t), filled,style=""):
!images/PLOT2DLasigi

```

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):
!images/PLOT2DLasigi

```

Berikut adalah himpunan nilai maksimal dari empat kondisi linier yang kurang dari atau sama dengan 3. Ini adalah  $A[k].vlt;=3$  untuk semua baris A. Untuk mendapatkan sudut yang bagus, kita menggunakan n yang relatif besar.

```

A=[2,1;1,2;-1,0;0,-1];
function f(x,y) := max([x,y].A');
plot2d("f",r=4,level=[0;3],color=green,n=111):
!images/PLOT2DLasigi

```

Poin utama dari bahasa matriks adalah memungkinkan pembuatan tabel fungsi dengan mudah.

```

t=linspace(0,2pi,1000); x=cos(3t); y=sin(4t);

```

Kami sekarang memiliki nilai vektor x dan y. plot2d() dapat memplot nilai-nilai ini sebagai kurva yang menghubungkan titik-titik tersebut. Plotnya bisa diisi. Dalam hal ini ini menghasilkan hasil yang bagus karena aturan belitan, yang digunakan untuk isi.

```

plot2d(x,y,igrid,iframe, filled):
!images/PLOT2DLasigi

```

Vektor interval diplot terhadap nilai x sebagai wilayah terisi antara nilai interval yang lebih rendah dan lebih tinggi. Hal ini dapat berguna untuk memplot kesalahan perhitungan. Tapi itu bisa juga dapat digunakan untuk memplot kesalahan statistik.

```

t=0:0.1:1; ... plot2d(t,interval(t-random(size(t)),t+random(size(t))),style="—");
... plot2d(t,t,add=true):
!images/PLOT2DLasigi

```

Jika x adalah vektor yang diurutkan, dan y adalah vektor interval, maka plot2d akan memplot rentang interval yang terisi pada bidang. Gaya isian sama dengan gaya poligon.

```

t=-1:0.01:1; x= t-0.01,t+0.01 ; y=x3 - x;
plot2d(t,y):
!images/PLOT2DLasigi

```

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 := "2x2 + xy + 3y4 + y"; //define an expression f(x,y)
plot2d(expr,level=[0;1],style="—",color=blue): // 0 := f(x,y) := 1
!images/PLOT2DLasigi

```

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 = "/") :
```

```
!images/PLOT2D_Lasigi
```

```
plot2d("cos(x)", "sin(x)^3", xmin = 0, xmax = 2pi, filled, style = "/") :
```

```
!images/PLOT2D_Lasigi
```

Grafik Fungsi Parametrik

Nilai x tidak perlu diurutkan. (x,y) hanya menggambarkan sebuah kurva.

Jika x diurutkan, kurva tersebut merupakan grafik suatu fungsi.

Dalam contoh berikut, kita memplot spiral

$$\gamma(t) = t \cdot (\cos(2\pi t), \sin(2\pi t))$$

Kita perlu menggunakan banyak titik untuk tampilan yang halus atau fungsi `adaptive()` untuk mengevaluasi ekspresi (lihat fungsi `adaptive()` untuk lebih jelasnya).

```
t=linspace(0,1,1000); ... plot2d(tcos(2pit),tsin(2pit),r=1):
```

```
!images/PLOT2D_Lasigi
```

Sebagai alternatif, dimungkinkan untuk menggunakan dua ekspresi untuk kurva. Berikut ini plot kurva yang sama seperti di atas.

```
plot2d("xcos(2pix)", "xsin(2pix)", xmin=0, xmax=1, r=1):
```

```
!images/PLOT2D_Lasigi
```

```
t=linspace(0,1,1000); r=exp(-t); x=rcos(2pit); y=rsin(2pit);
```

```
plot2d(x,y,r=1):
```

```
!images/PLOT2D_Lasigi
```

Pada contoh berikutnya, kita memplot kurvanya

$$\gamma(t) = (r(t) \cos(t), r(t) \sin(t))$$

dengan

$$r(t) = 1 + \sin(3t)2.$$

```
t=linspace(0,2pi,1000); r=1+sin(3t)/2; x=rcos(t); y=rsin(t); ... plot2d(x,y, filled, fillcolor=red, style="/", r=1.
```

```
!images/PLOT2D_Lasigi
```

Menggambar Grafik Bilangan Kompleks

Serangkaian bilangan kompleks juga dapat diplot. Kemudian titik-titik grid akan dihubungkan. Jika sejumlah garis kisi ditentukan (atau vektor garis kisi 1x2) dalam argumen `cgrid`, hanya garis kisi tersebut yang terlihat.

Matriks bilangan kompleks secara otomatis akan diplot sebagai kisi-kisi pada bidang kompleks.

Pada contoh berikut, kita memplot gambar lingkaran satuan di bawah fungsi eksponensial. Parameter `cgrid` menyembunyikan beberapa kurva grid.

```
aspect(); r=linspace(0,1,50); a=linspace(0,2pi,80); z=rexp(1a);... plot2d(z,a=-1.25,b=1.25,c=-1.25,d=1.25,cgrid=10):
```

```
!images/PLOT2D_Lasigi
```



```

aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)'; z=rexp(Ia);
plot2d(exp(z),cgrid=[40,10]):
!images/PLOT2DLasigi
r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=rexp(Ia);
plot2d(exp(z), points, add):
!images/PLOT2DLasigi

```

Vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian nyata dan bagian imajiner.

Dalam contoh, kita memplot lingkaran satuan dengan

$$\gamma(t) = e^{it}$$

```

t=linspace(0,2pi,1000); ... plot2d(exp(1t)+exp(4It),r=2):
!images/PLOT2DLasigi

```

Plot Statistik

Ada banyak fungsi yang dikhususkan pada plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

Jumlah kumulatif dari nilai terdistribusi normal 0-1 menghasilkan jalan acak.

```

plot2d(cumsum(randnormal(1,1000))):
!images/PLOT2DLasigi

```

Penggunaan dua baris menunjukkan jalan dalam dua dimensi.

```

X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]):
!images/PLOT2DLasigi

```

columnsplot(cumsum(random(10)),style="/",color=blue):

```

!images/PLOT2DLasigi

```

Itu juga dapat menampilkan string sebagai label.

```

months=["Jan","Feb","Mar","Apr","May","Jun", ... "Jul","Aug","Sep","Oct","Nov","Dec"];
values=[10,12,12,18,22,28,30,26,22,18,12,8];
columnsplot(values,lab=months,color=red,style="-");
title("Temperature");
!images/PLOT2DLasigi

```

```

k=0:10;
plot2d(k,bin(10,k), bar):
!images/PLOT2DLasigi

```

```

plot2d(k,bin(10,k)); plot2d(k,bin(10,k), points, add):
!images/PLOT2DLasigi

```

```

plot2d(normal(1000),normal(1000), points,grid=6,style=".."):
!images/PLOT2DLasigi

```

```

plot2d(normal(1,1000), distribution,style="O"):
!images/PLOT2DLasigi

```

```

plot2d("qnormal",0,5;2.5,0.5, filled):
!images/PLOT2DLasigi

```

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

```

```
!images/PLOT2DLasigi
```

Atau Anda dapat menghitung distribusi dari data dan memplot hasilnya dengan `gt;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):
```

```
!images/PLOT2DLasigi
```

Fungsi `statplot()` mengatur gaya dengan string sederhana.

```
statplot(1:10,cumsum(random(10)),"b"):
```

```
!images/PLOT2DLasigi
```

```
n=10; i=0:n; ... plot2d(i,bin(n,i)/2n, a = 0, b = 10, c = 0, d = 0.3); ...plot2d(i,bin(n,i)/2n, points =
true, style = "ow", add = true, color = blue) :
```

```
!images/PLOT2DLasigi
```

Selain itu, data dapat diplot sebagai batang. Dalam hal ini, `x` harus diurutkan dan satu elemen lebih panjang dari `y`. Batangnya akan memanjang dari `x[i]` hingga `x[i+1]` dengan nilai `y[i]`. Jika `x` berukuran sama dengan `y`, maka `x` akan diperpanjang satu elemen dengan spasi terakhir.

Gaya isian dapat digunakan seperti di atas.

```
n=10; k=bin(n,0:n); ... plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray):
```

```
!images/PLOT2DLasigi
```

Data untuk plot bar (`bar=1`) dan histogram (`histogram=1`) dapat diberikan secara eksplisit dalam `xv` dan `yv`, atau dapat dihitung dari distribusi empiris dalam `xv` dengan `gt;distribusi` (atau `distribusi=n`). Histogram nilai `xv` akan dihitung secara otomatis dengan `gt;histogram`. Jika `gt;even` ditentukan, nilai `xv` akan dihitung dalam interval bilangan bulat.

```
plot2d(normal(10000),distribution=50):
```

```
!images/PLOT2DLasigi
```

```
k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m, bar):
```

```
!images/PLOT2DLasigi
```

```
columnplot(m,k):
```

```
!images/PLOT2DLasigi
```

```
plot2d(random(600)6,histogram=6):
```

```
!images/PLOT2DLasigi
```

Untuk distribusi, terdapat parameter `distribution=n`, yang menghitung nilai secara otomatis dan mencetak distribusi relatif dengan `n` sub-interval.

```
plot2d(normal(1,1000),distribution=10,style="
/"):
!images/PLOT2DLasigi
```

Dengan parameter `even=true`, ini akan menggunakan interval bilangan bulat.

```
plot2d(intrandom(1,1000,10),distribution=10,even=true):
```

```
!images/PLOT2DLasigi
```

Perhatikan bahwa ada banyak plot statistik yang mungkin berguna. Silahkan lihat tutorial tentang statistik.

```
columnplot(getmultiplicities(1:6,intrandom(1,6000,6))):
```

```
!images/PLOT2DLasigi
```

```
plot2d(normal(1,1000), distribution); ... plot2d("qnormal(x)", color=red, thickness=2, add):
!images/PLOT2D_Lasigi
```

Ada juga banyak plot khusus untuk statistik. Plot kotak menunjukkan kuartil distribusi ini dan banyak outlier. Menurut definisinya, outlier dalam plot kotak adalah data yang melebihi 1,5 kali rentang 50

```
M=normal(5,1000); boxplot(quartiles(M)):
!images/PLOT2D_Lasigi
```

Fungsi Implisit

Fungsi implisit adalah jenis fungsi dimana variabel dependen tidak dapat dipisahkan secara eksplisit dari variabel independen. Bentuk dari fungsi implisit sendiri adalah  $f(x,y)=0$ , dimana variabel, koefisien, dan konstanta sebagai persamaan di sisi kiri, dan disamakan dengan nol.

Contoh dari fungsi implisit misalnya persamaan lingkaran dan persamaan elips.

Plot implisit menunjukkan penyelesaian garis level  $f(x,y)=level$ , dengan "level" dapat berupa nilai tunggal atau vektor nilai. Jika level = "auto", akan ada garis level nc, yang akan tersebar antara fungsi minimum dan maksimum secara merata.

Perintah `gt;hue` untuk membuat Warna yang lebih gelap atau lebih terang untuk menunjukkan nilai fungsi. Untuk fungsi implisit, `xv` harus berupa fungsi atau ekspresi parameter `x` dan `y`, atau alternatifnya, `xv` dapat berupa matriks nilai.

Euler dapat menandai garis level

$$f(x, y) = c$$

dari fungsi apa pun.

Untuk menggambar himpunan  $f(x,y)=c$  untuk satu atau lebih konstanta `c`, Anda dapat menggunakan `plot2d()` dengan plot implisitnya pada bidang. Parameter `c` adalah `level=c`, dimana `c` dapat berupa vektor garis level. Selain itu, skema warna dapat digambar di latar belakang untuk menunjukkan nilai fungsi setiap titik dalam plot. Parameter "n" menentukan kehalusan plot.

```
aspect();
plot2d("x^2 + y^2 - xy - x", level = 0, r = 1.5, contourcolor = red) :
!images/PLOT2D_Lasigi
expr := "2x^2 + xy + 3y^4 + y"; //define an expression f(x, y)
plot2d(expr, level=0): // Solutions of f(x,y)=0
!images/PLOT2D_Lasigi
plot2d(expr, level=0:0.5:20, hue, contourcolor=white, n=100): // nice
!images/PLOT2D_Lasigi
plot2d(expr, level=0:0.5:20, hue, spectral, n=200, grid=4): // nicer
!images/PLOT2D_Lasigi
```

Ini juga berfungsi untuk plot data. Namun Anda harus menentukan rentangnya untuk label sumbu.

```
x=-2:0.05:1; y=x'; z=expr(x,y);
plot2d(z, level=0, a=-1, b=2, c=-2, d=1, hue):
```

```
!images/PLOT2D_Lasigi
plot2d("x3 - y2", hue, spectral, contour) :
!images/PLOT2D_Lasigi
```

Perintah contour adalah untuk menentukan bahwa grafik yang dihasilkan adalah grafik kontur. grafik kontur menunjukkan garis-garis yang menghubungkan titik-titik dengan nilai fungsi yang sama, sehingga membuat visualisasi bentuk permukaan fungsi.

```
plot2d("x3 - y2", level = 0, contourwidth = 3, contourcolor = red, add) :
!images/PLOT2D_Lasigi
z=z+normal(size(z))0.2;
plot2d(z,level=0.5,a=-1,b=2,c=-2,d=1):
!images/PLOT2D_Lasigi
```

Perintah ini untuk menambahkan noise atau gangguan acak pada fungsi z. nilai yang dikalikan adalah standar deviasi atau besaran gangguan yang diberikan. semakin besar nilai yang dimasukkan maka semakin besar pula gangguan pada fungsi.

```
plot2d(expr,level=[0:0.2:5;0.05:0.2:5.05],color=lightgray):
!images/PLOT2D_Lasigi
plot2d("x2 + y3 + xy", level = 1, r = 4, n = 100) :
!images/PLOT2D_Lasigi
plot2d("x2 + 2y2 - xy", level = 0 : 0.1 : 10, n = 100, contourcolor =
white, hue) :
!images/PLOT2D_Lasigi
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 i= f(x,y) i= 1
!images/PLOT2D_Lasigi
```

Plot implisit juga dapat menunjukkan rentang level. Maka level harus berupa matriks interval level 2xn, di mana baris pertama berisi awal dan baris kedua berisi akhir setiap interval. Alternatifnya, vektor baris sederhana dapat digunakan untuk level, dan parameter dl memperluas nilai level ke interval.

```
plot2d("x4 + y4", level = [0; 1], r = 1.5, color = blue, style = "/" ) :
!images/PLOT2D_Lasigi
plot2d("x2 + y3 + xy", level = [0, 2, 4; 1, 3, 5], style = " / ", r = 2, n = 100) :
!images/PLOT2D_Lasigi
plot2d("x2 + y3 + xy", level = -10 : 20, dl = 0.1, r = 5, style = " - ", n =
100) :
!images/PLOT2D_Lasigi
plot2d("sin(x)cos(y)", levels,r=2pi, hue,n=100):
!images/PLOT2D_Lasigi
```

Dimungkinkan juga untuk menandai suatu wilayah

$$a \leq f(x, y) \leq b.$$

Hal ini dilakukan dengan menambahkan level dengan dua baris.

```
plot2d("(x^2+y^2-1)^3-x^2y^3", r = 6, ...style = "", color = red, < outline, ...level =
[-2; 0], n = 100) :
!images/PLOT2D_Lasigi
plot2d("4x^2-4.5", -0.75, 0.75, thickness = 3, add); plot2d([0, 0], [3, 5], thickness =
3, add) :
!images/PLOT2D_Lasigi
plot2d("(x^2+y^2-1)^3-x^2y^3", r = 6, ...style = "", color = red, < outline, ...level =
[-1; 0], n = 100) :
!images/PLOT2D_Lasigi
plot2d([-3.5,-3.5],[-1.1,1.1],thickness=3, add); plot2d("4(x-4)^2-1", 3.3, 4.7, thickness =
3, add) :
!images/PLOT2D_Lasigi
```

Dimungkinkan untuk menentukan level tertentu. Misalnya, kita dapat memplot solusi persamaan seperti

$$x^3 - xy + x^2y^2 = 6$$

```
plot2d("x^3 - xy + x^2y^2", r = 6, level = 1, n = 100) :
!images/PLOT2D_Lasigi
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 poly-
gon([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); hold-
ing(h); window(w); endfunction i/prej. Tidak ada tanda centang kotak atau
sumbu di sini. Selain itu, kami menggunakan jendela penuh untuk plotnya.
```

Kami memanggil reset sebelum kami menguji plot ini untuk mengembalikan default grafis. Ini tidak perlu dilakukan jika Anda yakin plot Anda berhasil.

```
reset; starplot1(normal(1,10)+2,color=red,lab=1:10):
!images/PLOT2D_Lasigi
* Starplot1 adalah perintah untuk menggambar plot berbentuk bintang. *
Ini adalah fungsi untuk menghasilkan vektor acak yang mengikuti * distribusi
normal dengan rata-rata 1 dan deviasi standar 10. Fungsi * ini menghasilkan
data yang terdistribusi normal dengan nilai-nilai * acak. * +5 adalah untuk
menambahkan 5 ke setiap elemen darivektor yang * terdistribusi normal.
```

Terkadang, Anda mungkin ingin merencanakan sesuatu yang plot2d tidak bisa lakukan, tapi hampir.

Dalam fungsi berikut, kita membuat plot impuls logaritmik. plot2d dapat melakukan plot logaritmik, tetapi tidak untuk batang impuls.

```
function logimpulseplot1 (x,y) ...
```

```

x0,y0=makeimpulse(x,log(y)/log(10)); plot2d(x0,y0,i,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); end if; end; holding(h); endfunction < /pre > Marikitauidengannilaiyangterdistribusi
aspect(1.5); x=1:10; y=-log(random(size(x)))200; ... logimpulseplot1(x,y):
!images/PLOT2D_Lasigi

```

Mari kita menganimasikan kurva 2D menggunakan plot langsung. Perintah plot(x,y) hanya memplot kurva ke dalam jendela plot. setplot(a,b,c,d) menyetel jendela ini.

Fungsi wait(0) memaksa plot muncul di jendela grafis. Jika tidak, pengundian ulang akan dilakukan dalam interval waktu yang jarang.

```

function animliss (n,m) ...
t=linspace(0,2pi,500); f=0; c=framecolor(0); l=linewidth(2); setplot(-1,1,-1,1); repeat clg; plot(sin(n*t),cos(m*t+f)); wait(0); if testkey() then break; end if; f=f+0.02; end; framecolor(c); linewidth(1); endfunction i/pre; Tekan tombol apa saja untuk menghentikan animasi ini.

```

```
animliss(5,5); // lihat hasilnya, jika sudah puas, tekan ENTER
```

```
User interrupted! Try "trace errors" to inspect local variables after errors.
animliss: if testkey() then break; end if; Error in: animliss(5,5); // lihat hasilnya, jika sudah puas, tekan ENTER ...

```

Plot Logaritmik

EMT menggunakan parameter "logplot" untuk skala logaritmik.

Plot logaritma dapat diplot menggunakan skala logaritma di y dengan logplot=1, atau menggunakan skala logaritma di x dan y dengan logplot=2, atau di x dengan logplot=3.

- logplot=1: y-logarithmic - logplot=2: x-y-logarithmic - logplot=3: x-logarithmic

```

plot2d("exp(x^3 - x)x^2", 1, 5, logplot = 1) :
!images/PLOT2D_Lasigi
plot2d("exp(x+sin(x))", 0, 100, logplot=1):
!images/PLOT2D_Lasigi
plot2d("exp(x+sin(x))", 10, 100, logplot=2):
!images/PLOT2D_Lasigi
plot2d("gamma(x)", 1, 10, logplot=1):
!images/PLOT2D_Lasigi
plot2d("log(x(2+sin(x/100)))", 10, 1000, logplot=3):
!images/PLOT2D_Lasigi

```

Ini juga berfungsi dengan plot data.

```

x=10(1 : 20); y = x^2 - x;
plot2d(x,y,logplot=2):
!images/PLOT2D_Lasigi

```

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

"", "lt;gt;", ".", "..", "...",

"\*", "+", "\_", "-", "o"

"", "lt;gt;", "o" (filled shapes)

"w", "lt;gt;w", "ow" (non-transparent)

For lines use

\_, "\_", "-.", ".", "-.-", "-.-", "-gt;"

For filled polygons or bar plots use

", "O", "O", "/", ";", "

+", "\_", "\_", "t"

points : plot single points instead of line segments

addpoints : if true, plots line segments and points

add : add the plot to the existing plot  
 user : enable user interaction for functions  
 delta : step size for user interaction  
 bar : bar plot (x are the interval bounds, y the interval values)  
 histogram : plots the frequencies of x in n subintervals  
 distribution=n : plots the distribution of x with n subintervals  
 even : use inter values for automatic histograms.  
 steps : plots the function as a step function (steps=1,2)  
 adaptive : use adaptive plots (n is the minimal number of steps)  
 level : plot level lines of an implicit function of two variables  
 outline : draws boundary of level ranges.  
 If the level value is a 2xn matrix, ranges of levels will be drawn in the color using the given fill style. If outline is true, it will be drawn in the contour color. Using this feature, regions of f(x,y) between limits can be marked.  
 hue : add hue color to the level plot to indicate the function value  
 contour : Use level plot with automatic levels  
 nc : number of automatic level lines  
 title : plot title (default "")  
 xl, yl : labels for the x- and y-axis  
 smaller : if gt;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 `gt;user`, you get the same user interaction as in `plot2d`. The range will be set before each call to your function.  
 maps : map expressions (0 is faster), functions are always mapped.  
 contourcolor : color of contour lines  
 contourwidth : width of contour lines  
 clipping : toggles the clipping (default is true)  
 title :  
 This can be used to describe the plot. The title will appear above the plot. Moreover, a label for the x and y axis can be added with `xl="string"` or `yl="string"`. Other labels can be added with the functions `label()` or `labelbox()`. The title can be a unicode



string or an image of a Latex formula.

cgrid :

Determines the number of grid lines for plots of complex grids. Should be a divisor of the the matrix size minus 1 (number of subintervals). cgrid can be a vector [cx,cy].

Overview

The function can plot

- \* expressions, call collections or functions of one variable,
- \* parametric curves,
- \* x data against y data,
- \* implicit functions,
- \* bar plots,
- \* complex grids,
- \* polygons.

If a function or expression for xv is given, plot2d() will compute values in the given range using the function or expression. The expression must be an expression in the variable x. The range must be defined in the parameters a and b unless the default range [-2,2] should be used. The y-range will be computed automatically, unless c and d are specified, or a radius r, which yields the range [-r,r] for x and y. For plots of functions, plot2d will use an adaptive evaluation of the function by default. To speed up the plot for complicated functions, switch this off with lt;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 lt;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.

## 4 Plot 3D

PLOT3D<sub>LasigiYatindraJago</sub>23030630008 — — —

NAMA : Lasigi Yatindra Jago

KELAS : MATEMATIKA B

NIM : 23030630008

—

Menggambar Plot 3D dengan EMT

Ini adalah pengenalan plot 3D di Euler. Kita membutuhkan plot 3D untuk memvisualisasikan fungsi dari dua variabel.

Euler menggambar fungsi tersebut menggunakan algoritma pengurutan untuk menyembunyikan bagian di latar belakang. Secara umum, Euler menggunakan proyeksi pusat. Standarnya adalah dari kuadran x-y positif menuju titik asal  $x=y=z=0$ , tetapi sudut= $0^\circ$  terlihat dari arah sumbu y. Sudut pandang dan tinggi dapat diubah.

Euler dapat merencanakan

- \* permukaan dengan bayangan dan garis level atau rentang level,
- \* awan poin,
- \* kurva parametrik,
- \* permukaan implisit.

Plot 3D dari suatu fungsi menggunakan plot3d. Cara termudah adalah dengan memplot ekspresi dalam x dan y. Parameter r mengatur kisaran plot di sekitar (0,0).

aspect(1.5); plot3d("x<sup>2</sup> + sin(y)", r = pi) :

![images/PLOT3D\_Lasigi

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<sup>2</sup> + y<sup>2</sup>") :

![images/PLOT3D\_Lasigi

Interaksi pengguna dimungkinkan dengan gt;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<sup>2</sup>+y<sup>2</sup>)", user,...title = "Turnwiththevectorkeys(pressreturntofinish)") :

![images/PLOT3D\_Lasigi

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 `1t;fscale`).  
`skala`: angka atau vektor  $1 \times 2$  untuk skala ke arah  $x$  dan  $y$ .  
`bingkai`: jenis bingkai (default 1).  
`plot3d("exp(-(x2+y2)/5)", r = 10, n = 80, fscale = 4, scale = 1.2, frame = 3) :`  
`!images/PLOT3D_Lasigi`  
 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]`  
 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("x2 + y2", distance = 3, zoom = 2, angle = 0, height = 0) :`  
`!images/PLOT3D_Lasigi`  
 Plot terlihat selalu ke pusat kubus plot. Anda dapat memindahkan pusat dengan parameter tengah.  
`plot3d("x4 + y2", a = 0, b = 1, c = -1, d = 1, angle = -20, height = 20, ... center = [0.4, 0, 0], zoom = 5) :`  
`!images/PLOT3D_Lasigi`  
 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("5exp(-x2 - y2)", r = 2, < fscale, < scale, distance = 13, height = 50, ... center = [0, 0, -2], frame = 3) :`  
`!images/PLOT3D_Lasigi`  
 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/(x2 + y2 + 1)", r = 5, polar, ... fscale = 2, hue, n = 100, zoom = 4, contour, color = gray) :`  
`!images/PLOT3D_Lasigi`  
`function f(r) := exp(-r/2)cos(r); ... plot3d("f(x2 + y2)", polar, scale = [1, 1, 0.4], r = 2pi, frame = 3, zoom = 4) :`  
`!images/PLOT3D_Lasigi`  
 Rotasi parameter memutar fungsi dalam  $x$  di sekitar sumbu  $x$ .

```
* rotate=1: Menggunakan sumbu x
* rotate=2: Menggunakan sumbu z
plot3d("x2 + 1", a = -1, b = 1, rotate = true, grid = 5) :
```

```
!images/PLOT3DLasigi
```

Berikut adalah plot dengan tiga fungsi.

```
plot3d("x", "x2 + y2", "y", r = 2, zoom = 3.5, frame = 3) :
!images/PLOT3DLasigi
```

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.

-gt; hue: Menyalakan bayangan cahaya alih-alih kabel.

-gt; kontur: Memplot garis kontur otomatis pada plot.

-gt; 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(-x2-y2)", r = 2, n = 100, level = "thin", ...contour, spectral, fscale = 1, scale = 1.1, angle = 45, height = 20) :
```

```
!images/PLOT3DLasigi
```

```
plot3d("exp(xy)", angle=100°, contour,color=green):
```

```
!images/PLOT3DLasigi
```

Bayangan default menggunakan warna abu-abu. Tetapi rentang warna spektral juga tersedia.

-gt; spektral: Menggunakan skema spektral default

-gt; color=...: Menggunakan warna khusus atau skema spektral

Untuk plot berikut, kami menggunakan skema spektral default dan menambahkan jumlah titik untuk mendapatkan tampilan yang sangat halus.

```
plot3d("x2 + y2", spectral, contour, n = 100) :
```

```
!images/PLOT3DLasigi
```

Alih-alih garis level otomatis, kita juga dapat mengatur nilai garis level. Ini akan menghasilkan garis level tipis alih-alih rentang level.

```
plot3d("x2-y2", 0, 1, 0, 1, angle = 220, level = -1 : 0.2 : 1, color = redgreen) :
```

```
!images/PLOT3DLasigi
```

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("x2 + y3", level = [-0.1, 0.9; 0, 1], ... spectral, angle = 30, grid = 10, contourcolor = gray) :
```

```
!images/PLOT3DLasigi
```

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("xy - yx", level = 0, a = 0, b = 6, c = 0, d = 6, contourcolor =
red, n = 100) :
```

```
!images/PLOT3D_Lasigi
```

Dimungkinkan untuk menunjukkan bidang kontur di bawah plot. Warna dan jarak ke plot dapat ditentukan.

```
plot3d("x2 + y4", cp, cpcolor = green, cpdelta = 0.2) :
```

```
!images/PLOT3D_Lasigi
```

Berikut ini beberapa gaya lainnya. Kami selalu mematikan bingkai, dan menggunakan berbagai skema warna untuk plot dan kisi-kisi.

```
figure(2,2); ... Dimungkinkan untuk menunjukkan bidang kontur di bawah
plot. Warna dan jarak ke plot dapat ditentukan.
```

```
expr="y3-x2";...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) :
```

```
!images/PLOT3D_Lasigi
```

Ada beberapa skema spektral lainnya, bernomor dari 1 hingga 9. Tetapi Anda juga dapat menggunakan warna=nilai, di mana nilai

- \* spektral: untuk rentang dari biru ke merah

- \* putih: untuk rentang yang lebih redup \* -kuningbiru, ungu hijau, birukuning, hijaumerah

- \* birukuning, hijau ungu, kuning biru, merah hijau

```
figure(3,3); ... for i=1:9; ... figure(i); plot3d("x2 + y2", spectral =
i, contour, cp, < frame, zoom = 4);... end;... figure(0) :
```

```
!images/PLOT3D_Lasigi
```

Sumber cahaya dapat diubah dengan l dan tombol kursor selama interaksi pengguna. Itu juga dapat diatur dengan parameter.

- \* cahaya: arah untuk cahaya

- \* amb: cahaya sekitar antara 0 dan 1

Perhatikan bahwa program tidak membuat perbedaan antara sisi plot. Tidak ada bayangan. Untuk ini, Anda perlu Povray.

```
plot3d("-x2-y2",...hue = true, light = [0, 1, 1], amb = 0, user = true, ...title =
"Presslandcursorkeys(returntoexit)") :
```

```
!images/PLOT3D_Lasigi
```

Parameter warna mengubah warna permukaan. Warna garis level juga dapat diubah.

```
plot3d("-x2-y2", color = rgb(0.2, 0.2, 0), hue = true, frame = false, ...zoom =
3, contourcolor = red, level = -2 : 0.1 : 1, dl = 0.01) :
```

```
!images/PLOT3D_Lasigi
```

Warna 0 memberikan efek pelangi khusus.

```
plot3d("x2/(x2 + y2 + 1)", color = 0, hue = true, grid = 10) :
```

```
!images/PLOT3D_Lasigi
```

Permukaannya juga bisa transparan.

```
plot3d("x^2 + y^2", transparent, grid = 10, wirecolor = red) :
```

```
!images/PLOT3D_Lasigi
```

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

lateks:  $f(x,y,z) = 0$

dapat divisualisasikan dalam potongan sejajar dengan bidang x-y-, x-z- dan y-z.

\* implisit=1: potong sejajar dengan bidang y-z

\* implisit=2: potong sejajar dengan bidang x-z

\* implisit=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 + zy - 1", r = 5, implicit = 3) :
```

```
!images/PLOT3D_Lasigi
```

```
plot3d("x^2 + y^2 + 4xz + z^3", implicit, r = 2, zoom = 2.5) :
```

```
!images/PLOT3D_Lasigi
```

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,iframe,grid=[10,20], ...
values=s,contourcolor=red,level=[90°-24°;90°-22°], ... scale=1.4,height=50°):
```

```
!images/PLOT3D_Lasigi
```

Berikut adalah contoh, yang merupakan grafik fungsi.

```
t=-1:0.1:1; s=(-1:0.1:1)'; plot3d(t,s,ts,grid=10):
```

```
!images/PLOT3D_Lasigi
```

Namun, kita bisa membuat segala macam permukaan. Berikut adalah permukaan yang sama dengan fungsi

$$x = yz$$

plot3d(ts,t,s,angle=180°,grid=10):

! [images/PLOT3D<sub>Lasigi</sub>

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(4t)+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):

! [images/PLOT3D<sub>Lasigi</sub>

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=" "):

! [images/PLOT3D<sub>Lasigi</sub>

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):

! [images/PLOT3D<sub>Lasigi</sub>

t=linspace(0,4pi,1000); plot3d(cos(t),sin(t),t/2pi, wire, ... linewidth=3,wirecolor=blue):

! [images/PLOT3D<sub>Lasigi</sub>

X=cumsum(normal(3,100)); ... plot3d(X[1],X[2],X[3], anaglyph, wire):

! [images/PLOT3D<sub>Lasigi</sub>

EMT juga dapat memplot dalam mode anaglyph. Untuk melihat plot seperti itu, Anda memerlukan kacamata merah/sian.

plot3d("x<sup>2</sup> + y<sup>3</sup>", anaglyph, contour, angle = 30) :

! [images/PLOT3D<sub>Lasigi</sub>

Seringkali, skema warna spektral digunakan untuk plot. Ini menekankan ketinggian fungsi.

plot3d("x<sup>2</sup>y<sup>3</sup> - y", spectral, contour, zoom = 3.2) :

! [images/PLOT3D<sub>Lasigi</sub>

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(0,2pi,50)'; ... X=(3+ucos(v/2))cos(v);
Y=(3+ucos(v/2))sin(v); Z=usin(v/2); ... plot3d(X,Y,Z, anaglyph,iframe, wire,scale=2.3):
!images/PLOT3DLasigi

```

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+.25sin(3 ×
v))+cos(u))cos(2v); ... y:=(4(1+.25sin(3v))+cos(u))sin(2v); ... z=sin(u)+2 ×
cos(3v); ... plot3d(x,y,z,frame=0,scale=1.5,hue=1,light=[1,0,-1],zoom=2.8, anaglyph):
!images/PLOT3DLasigi

```

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=x2+y2; ...xa = (x|1.1)-0.05; ya = (y|1.1)-0.05; ...plot3d(xa, ya, z, bar =
true) :

```

```
!images/PLOT3DLasigi
```

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):

```

```
!images/PLOT3DLasigi
```

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 gt;zscale. Ini dapat dikombinasikan dengan faktor penskalaan individu yang diterapkan sebagai tambahan.

```

i=1:20; j=i'; ... plot3d(ij2+100normal(20,20), zscale, scale = [1, 1, 1.5], angle =
-40, zoom = 1.8) :

```

```
!images/PLOT3DLasigi
```

```

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]):

```

```
!images/PLOT3DLasigi
```

Permukaan Benda Putar

```

plot2d("(x2+y2-1)3-x2y3", r = 1.3, ...style = "", color = red, < outline, ...level =
[-2; 0], n = 100) :

```

```
!images/PLOT3DLasigi
```

```

ekspresi = (x2 + y2 - 1)3 - x2y3;ekspresi

```

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

Kami ingin memutar kurva jantung di sekitar sumbu y. Berikut adalah ungkapan, yang mendefinisikan hati:

```

lateks: f(x,y)=(x2 + y2 - 1)3 - x2.y3.

```

Selanjutnya kita atur

```

lateks: x=r.cos(a), y=r.sin(a).

```



function fr(r,a) = ekspresi with [x=rcos(a),y=rsin(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(rcos(t)sin(s),rcos(t)cos(s),r *
sin(t), ... hue,i)frame,color=red,zoom=4,amb=0,max=0.7,grid=12,height=50°):
!images/PLOT3DLasigi
```

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=x2+y2; return(r+z2-1)3-r*z3; endfunction < /pre > 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) :
!images/PLOT3DLasigi
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 sukai.

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,ijscale,ijsframe,ijs hue, .. hues=0.5,ijs contour,color=orange)
h=holding(1); plot3d(x,y,(x2+y2)/2,< scale,< frame,> contour,> hue); holding(h); endfunction <
/pre > Sekarang framedplot() menyediakan frame, dan mengaturnya.
framedplot("myplot",[0.1,1,0,1,0,1],angle=-45°, ... center=[0,0,-0.7],zoom=6):
!images/PLOT3DLasigi
```

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=x2-y4;
function myplot (x,y,z) ... ijspre class="udf" ijs zoom(2); wi=fullwindow();
plotcontourplane(x,y,z,level="auto",ijscale); plot3d(x,y,z,ijs hue,ijscale,ijs add,color=white,level="thin");
window(wi); reset(); endfunction ijspre ijs myplot(x,y,z):
!images/PLOT3DLasigi
```

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 menjiwai plot.

Silakan pelajari sumber rotasi untuk melihat lebih detail.

```
function testplot () := plot3d("x2 + y3"); ... rotate("testplot"); testplot() :
!images/PLOT3DLasigi
```

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". <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:. 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:\Files-Ray\3.7.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("x2 + y2", zoom = 3);
```

```
!images/PLOT3D_Lasigi
```

Kita dapat membuat fungsi menjadi transparan dan menambahkan hasil akhir lainnya. Kami juga dapat menambahkan garis level ke plot fungsi.

```
pov3d("x2+y3", axiscolor = red, angle = 20, ...look = povlook(blue, 0.2), level = -1 : 0.5 : 1, zoom = 3.8);
```

```
!images/PLOT3D_Lasigi
```

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 + y2)((x+1)2 + y2)/40", r = 1.5, ... angle = -120, level = 1/40, dlevel = 0.005, light = [-1, 1, 1], height = 45, n = 50, ... < fscale, zoom = 3.8);
```

```
!images/PLOT3D_Lasigi
```

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) := x3 - 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]);
```

```
!images/PLOT3D_Lasigi
```

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=rcos(phi); y=rsin(phi); z=exp(-5r)cos(8pir)/3; ... pov3d(x,y,z, zoom=5, axis=0, add=povsphere([0,0,0.5], 0.1, povlook(green)), ... w=500, h=300);
```

```
!images/PLOT3D_Lasigi
```

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.

$$dx = \text{diff}([x, y, Z], x); dy = \text{diff}([x, y, Z], y);$$

Kami mendefinisikan normal sebagai produk silang dari turunan ini, dan mendefinisikan fungsi koordinat.

```
N = crossproduct(dx,dy); NX = N[1]; NY = N[2]; NZ = N[3]; N,
3 2 2 [- 2 x y , - 3 x y , 1]
```

Kami hanya menggunakan 25 poin.

```
x=-1:0.5:1; y=x';
pov3d(x,y,Z(x,y),angle=10°, ... xv=NX(x,y),yv=NY(x,y),zv=NZ(x,y),;shadow);
!images/PLOT3D_Lasigi
```

Berikut ini adalah simpul Trefoil yang dilakukan oleh A. Busser di Povray.

Ada versi yang ditingkatkan dari ini dalam contoh.

Lihat: ContohSimpul — 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(3y))+cos(x))cos(2y); ... Y = ((4+sin(3y))+cos(x))sin(2y);
... Z = sin(x)+2cos(3y);
```

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(-
```

Vektor normal adalah evaluasi dari ekspresi simbolik dn[i] untuk i=1,2,3.

Sintaks untuk ini adalah amp;"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));
!images/PLOT3D_Lasigi
```

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 =
rcos(t); z = rsin(t); ... writeln(povgrid(x,y,z,d = 0.02,dballs = 0.05)); ... povend();
!images/PLOT3D_Lasigi
```

Dengan povgrid(), kurva dimungkinkan.

```
povstart(center=[0,0,1],zoom=3.6); ... t=linspace(0,2,1000); r=exp(-t); ...
x=cos(2pi10t)r; y=sin(2pi10t)r; z=t; ... writeln(povgrid(x,y,z,povlook(red)));
... writeAxis(0,2,axis=3); ... povend();
!images/PLOT3D_Lasigi
```

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)); ...
```

Pertama kita mendefinisikan tiga silinder, dan menyimpannya dalam string di Euler.

Fungsi `povx()` dll. hanya mengembalikan vektor  $[1,0,0]$ , yang dapat digunakan sebagai pengingat.

```
c1
cylinder lt;-1,0,0gt;, lt;1,0,0gt;, 1 texture pigment color rgb lt;0.564706,0.0627451,0.0627451gt;
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)
texture pigment color rgbf lt;0.101961,0.2,0.301961,0.1gt; finish ambient
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;
!images/PLOT3DLasigi
```

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) ... i/pre class="udf" i 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 i/pre i povstart(fade=10,ishadow);
fractal(-1,-1,-1,2,4);
povend();
!images/PLOT3DLasigi
```

Perbedaan memungkinkan memotong satu objek dari yang lain. Seperti persimpangan, ada bagian dari objek CSG 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();
```

!images/PLOT3D<sub>L</sub>asigi

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();
```

!images/PLOT3D<sub>L</sub>asigi

Objek Jala

Dalam contoh ini, kami menunjukkan cara membuat objek mesh, dan menggambarinya 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);
```

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=xy; vx=-y; vy=-x; vz=1;
```

```
mesh=povtriangles(x,y,z,"",vx,vy,vz);
```

Sekarang kita tentukan dua cakram, yang akan berpotongan dengan permukaan.

```
cl=povdisc([0.5,0.5,0],[1,1,0],2); ... ll=povdisc([0,0,1/4],[0,0,1],2);
```

Tuliskan permukaan dikurangi kedua cakram.

```
writeln(povdifference(mesh,povunion([cl,ll],povlook(green)));
```

Write the two intersections.

```
writeln(povintersection([mesh,cl],povlook(red))); ... writeln(povintersection([mesh,ll],povlook(gray)));
```

Tulis titik pada maksimum.

```
writeln(povpoint([1/2,1/2,1/4],povlook(gray),size=2defaultpointsize));
```

Tambahkan sumbu dan selesaikan.

```
writeAxes(0,1,0,1,0,1,d=0.015); ... povend();
```

```
!images/PLOT3DLasigi
```

Anaglyph dalam Povray

Untuk menghasilkan anaglyph untuk kacamata merah/cyan, Povray harus dijalankan dua kali dari posisi kamera yang berbeda. Ini menghasilkan dua file Povray dan dua file PNG, yang dimuat dengan fungsi loadanaglyph().

Tentu saja, Anda membutuhkan kacamata merah/cyan untuk melihat contoh berikut dengan benar.

```
pov3d("-exp(-x2-y2)/2",r=2,height=45,anaglyph,...center=[0,0,0.5],zoom=3.5);
```

```
!images/PLOT3DLasigi
```

Jika Anda membuat scene dengan objek, Anda harus menempatkan pembuatan scene ke dalam suatu fungsi, dan menjalankannya dua kali dengan nilai yang berbeda untuk parameter anaglyph.

```
function myscene ...
```

```
s=povsphere(povc,1); cl=povcylinder(-povz,povz,0.5); clx=povobject(cl,rotate=xrotate(90°));
cly=povobject(cl,rotate=yrotate(90°)); c=povbox([-1,-1,0],1); un=povunion([cl,clx,cly,c]);
obj=povdifference(s,un,povlook(red)); writeln(obj); writeAxes(); endfunction
i/prej Fungsi povanaglyph() melakukan semua ini. Parameternya sama seperti pada povstart() dan povend() digabungkan.
```

```
povanaglyph("myscene",zoom=4.5);
```

```
!images/PLOT3DLasigi
```

Mendefinisikan objek

Interface dari Povray di Euler mengandung banyak objek, tidak terbatas pada itu. Kita bisa membuat objek kita sendiri, bisa menggabungkan objek dan menjadikannya objek baru

Kita akan mendemonstrasikan torus. Perintah pada Povray nya adalah "torus". Maka kita bisa mengembalikan stringnya menggunakan perintah ini dan parameternya. Perlu diingat bahwa torus akan selalu terletak di titik pusat (Origin Point)

```
function povdonat (r1,r2,look="") ...
```

```
return "torus "+r1+" "+r2+look+""; endfunction i/prej Ini adalah torus pertama kita
```

```
t1=povdonat(0.8,0.2)
```

```
torus 0.8,0.2
```

Kita akan menggunakan objek ini untuk membuat torus kedua, ditransaliskan dan di rotasi

```
t2=povobject(t1,rotate=xrotate(90°),translate=[0.8,0,0])
```

```
object torus 0.8,0.2 rotate 90 *x translate lt;0.8,0,0gt;
```

Sekarang kita letakkan objeknya. Kita akan menggunakan Phong Shading

```
povstart(center=[0.4,0,0],angle=0°,zoom=3.8,aspect=1.5); ... writeln(povobject(t1,povlook(green,phong=
... writeln(povobject(t2,povlook(green,phong=1))); ... gt;povend();
```

Ketik program Povray nya, bagaimanapun jika ada error, programnya tidak akan memperlihatkan error nya. Maka gunakan

```
gt;povend(lt;exit);
```

Jika tidak ada yang berhasil, maka window Povray akan tetap terbuka

```
povend(h=320,w=480);
![images/PLOT3D_Lasigi
Berikut adalah contoh lainnya
```

$$Ax \leq b, \quad x \geq 0, \quad c \cdot x \rightarrow \text{Max}.$$

Buktikan titik layak pdan optimum pada 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, kita cek apakah contoh ini memiliki solusi atau tidak

```
x=simplex(A,b,c, max, check)'
```

```
[0, 1, 0.5]
```

Kita definisikan 2 objek, pertama adalah bangun datar

$$a \cdot x \leq b$$

```
function oneplane (a,b,look="") ...
    return povplane(a,b,look) endfunction i/prej Definisikan perpotongan di se-
mua setengah ruang dan lingkaran
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 i/prej Sekarang kita plot
    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); ... Lingkaran di sekitar optimum
    writeln(povintersection([povsphere(x,0.5),povplane(c,c.x')], ... povlook(red,0.9)));
Error di arah optimum
    writeln(povarrow(x,c0.5,povlook(red)));
Kita akan menambahkan teks di layar. Teksnya berupa objek 3D. Kita harus
meletakkan teks berdasarkan sudut pandang kita.
    writeln(povtext("Linear Problem",[0,0.2,1.3],size=0.05,rotate=125°)); ...
povend();
    object text ttf "timrom.ttf" "Linear Problem" 0.04 ,0 texture pigment
color rgb lt;0.470588,0.470588,0.470588gt; finish ambient 0.2 scale 0.05
rotate 90 *x rotate 125 *z translate lt;0,0.2,1.3gt;
    ![images/PLOT3D_Lasigi
```

## 5 Kalkulus

MATERI KALKULUS Nama : Lasigi Yatindra Jago

NIM : 23030630008

Kelas: MATEMATIKA B

Kalkulus dengan EMT

Materi Kalkulus mencakup di antaranya:

\* Fungsi (fungsi aljabar, trigonometri, eksponensial, logaritma, \* komposisi fungsi)



- \* Limit Fungsi,
- \* Turunan Fungsi,
- \* Integral Tak Tentu,
- \* Integral Tentu dan Aplikasinya,
- \* Barisan dan Deret (kekonvergenan barisan dan deret).

EMT (bersama Maxima) dapat digunakan untuk melakukan semua perhitungan di dalam kalkulus, baik secara numerik maupun analitik (eksak).

Mendefinisikan Fungsi

Terdapat beberapa cara mendefinisikan fungsi pada EMT, yakni:

- \* Menggunakan format `nama_fungsi := rumus_fungsi(untuk_fungsi*numerik)`,
- \* Menggunakan format `nama_fungsi&= rumus_fungsi(untuk_fungsi*simbolik,namundapatdihitungsecaranumerik)`,
- \* Menggunakan format `nama_fungsi&= rumus_fungsi(untuk_fungsi*simbolikmurni,tidakdapatdihitunglangsung)`,
- \* Fungsi sebagai program EMT.

Setiap format harus diawali dengan perintah `function` (bukan sebagai ekspresi).

Berikut adalah beberapa contoh cara mendefinisikan fungsi.

Mendefinisikan fungsi:

$$f(x) = 2x^2 + e^{\sin(x)}.$$

```
function f(x) := 2x^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 ...
plot2d("f(x)",-1,1):
![images/KALKULUS_Lasigi
Berikutnya kita definisikan fungsi:
```

$$g(x) = \frac{\sqrt{x^2 - 3x}}{x + 1}.$$

```
function g(x) := sqrt(x^2 - 3x)/(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,tidakdapatdihitungolehfungsi numerik...
plot2d("g(x)",-1,1):
![images/KALKULUS_Lasigi
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
plot2d("h(x)",-1,1) :
![images/KALKULUSLasigi
f(0:10) // nilai-nilai 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]
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^3else return x^2endif;endfunction < /pre > 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):
![images/KALKULUSLasigi
function f(x) = 2E^x // fungsisimbolik
x 2 E
function g(x) = 3x+1
3 x + 1
function h(x) = f(g(x)) // komposisi fungsi
3 x + 1 2 E
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 beberapa nilainya, baik untuk satu nilai maupun vektor. Gambar grafik tersebut.

Juga, carilah fungsi beberapa (dua) variabel. Lakukan hal sama seperti di atas.

Jawab:

*A). FUNGSI I VARIABEL*

1. Fungsi 1  
function k(x) := x(x<sup>5</sup> + 3)<sup>3</sup>  
k(3), k(5), k(7)  
44660808 153027765760 3.3250729687e+13  
kmap(-3:3)  
[4.1472e+07, 48778, -8, 0, 64, 85750, 4.46608e+07]  
plot2d("k(x)"): !  
[images/KALKULUS<sub>Lasigi</sub>]
2. Fungsi 2  
function m(x) := (x)<sup>4</sup>/(3 - x<sup>2</sup>)  
m(2), m(-2), m(1)  
-16 -16 0.5  
mmap(-5:-5)  
-28.4090909091  
plot2d("m(x)"): !  
[images/KALKULUS<sub>Lasigi</sub>]
3. Fungsi 3  
function n(x) := 3x/(x+5)+2  
n(2), n(-1), n(-3), n(4)  
2.85714285714 1.25 -2.5 3.33333333333  
nmap(2:5)  
[2.85714, 3.125, 3.33333, 3.5]  
plot2d("n(x)"): !  
[images/KALKULUS<sub>Lasigi</sub>]
4. Fungsi 4  
function l(x) := 3x<sup>3</sup>/(x<sup>4</sup> - 3)  
l(5), l(4), l(3)  
0.602893890675 0.758893280632 1.03846153846  
lmap(5:8)  
[0.602894, 0.50116, 0.429108, 0.375275]  
plot2d("l(x)", -3, 3, -600, 600): !  
[images/KALKULUS<sub>Lasigi</sub>]
5. Fungsi 5  
function j(x) := (cos(x))sin(2x)  
j(pi), j(0), j(pi/3)  
0 0 0.433012701892  
jmap(0:3pi)  
[0, 0.491295, 0.314941, 0.276619, -0.646688, -0.154318, -0.515201, 0.746821,  
0.0418899, 0.684247]  
plot2d("j(x)"): !  
[images/KALKULUS<sub>Lasigi</sub>]
6. Fungsi 6

```

function o(x) := xsqrt(x+2)
o(3), o(5), o(7)
6.7082039325 13.2287565553 21
omap(3:12)
[6.7082, 9.79796, 13.2288, 16.9706, 21, 25.2982, 29.8496, 34.641, 39.6611,
44.8999]
plot2d("o(x)":
!images/KALKULUSLasigi

```

### B). FUNGSI 2 VARIABEL

#### 1. Fungsi 1

```

function a(x,y) ...
return x2 + y2 - 24endfunction < /pre > a(2,1), a(5,4), a(2,4)
-19 17 -4
amap(-2:2,3:3)
[-11, -14, -15, -14, -11]
aspect=1.5; plot3d("a(x,y)",a=-100,b=100,c=-80,d=80,angle=35°,height=30°,r=pi,n=100):
!images/KALKULUSLasigi

```

#### 2. Fungsi 2

```

function q(x,y) ...
return y2/(x2/3)endfunction < /pre > q(4,2), q(2,3), q(4,3)
0.75 6.75 1.6875
qmap(2:2,-2:2)
[3, 0.75, 0, 0.75, 3]
aspect=1.5; plot3d("q(x,y)",a=-100,b=100,c=-80,d=80,angle=35°,height=30°,r=pi,n=100):
!images/KALKULUSLasigi

```

#### 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(1/(2x-1), x, 0))
```

$$\lim_{x \rightarrow 0} \frac{1}{2x-1} = -1$$

```
showev('limit((x2 - 3x - 10)/(x - 5), x, 5))
```

$$\lim_{x \rightarrow 5} \frac{x^2 - 3x - 10}{x - 5} = 7$$

```
showev('limit(sin(x)/x, x, 0))
```

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

```
plot2d("sin(x)/x",-pi,pi):
!images/KALKULUSLasigi
showev('limit(sin(x^3)/x,x,0))
```

$$\lim_{x \rightarrow 0} \frac{\sin x^3}{x} = 0$$

```
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,minu))
```

$$\lim_{t \uparrow 2} t - \sqrt{2-t} = 2$$

```
showev('limit(t-sqrt(2-t),t,5,plus))//Perhatikanhasilnya
```

$$\lim_{t \downarrow 5} t - \sqrt{2-t} = 5 - \sqrt{3}i$$

```
plot2d("x-sqrt(2-x)",-2,5):
!images/KALKULUSLasigi
showev('limit((x^2-9)/(2x^2-5x-3),x,3))
```

$$\lim_{x \rightarrow 3} \frac{x^2 - 9}{2x^2 - 5x - 3} = \frac{6}{7}$$

```
showev('limit((1-cos(x))/x,x,0))
```

$$\lim_{x \rightarrow 0} \frac{1 - \cos x}{x} = 0$$

```
showev('limit((x^2+abs(x))/(x^2-abs(x)),x,0))
```

$$\lim_{x \rightarrow 0} \frac{|x| + x^2}{x^2 - |x|} = -1$$

```
showev('limit((1+1/x)^x,x,inf))
```

$$\lim_{x \rightarrow \infty} \left( \frac{1}{x} + 1 \right)^x = e$$

```
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))*

$$\lim_{x \rightarrow 0} (x+1)^{\frac{1}{x}} = e$$

*showev('limit((x/(x+k))^x,x,inf))*

$$\lim_{x \rightarrow \infty} \left( \frac{x}{x+k} \right)^x = e^{-k}$$

*showev('limit(sin(1/x),x,0))*

$$\lim_{x \rightarrow 0} \sin \left( \frac{1}{x} \right) = ind$$

*showev('limit(sin(1/x),x,inf))*

$$\lim_{x \rightarrow \infty} \sin \left( \frac{1}{x} \right) = 0$$

*plot2d("sin(1/x)",-5,5):*

*![images/KALKULUS\_Lasigi*

*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.

Jawab:

1. Fungsi 1

$$f(x) = 3x - 6 \frac{1}{x+2}$$

*showev('limit((3x-6)/(x+2),x,2))*

$$\lim_{x \rightarrow 2} \frac{3x-6}{x+2} = 0$$

*plot2d("(3x-6)/(x+2)",-2,3.5,-1,5):*

*![images/KALKULUS\_Lasigi*

2. Fungsi 2

$$f(x) = \cos 2x \frac{1}{\sin x - \cos x}$$

$$\text{showev('limit(cos(2x)/(sin(x) - cos (x)),x,0))}$$

$$\lim_{x \rightarrow 0} \frac{\cos(2x)}{\sin x - \cos x} = -1$$

$$\text{plot2d("cos(2x)/(sin(x) - cos (x))",-1,1):}$$

!images/KALKULUS<sub>L</sub>asigi

3. Fungsi 3

$$f(x) = 2x^2 - 2x + 5 \frac{1}{3x^2 + x - 6}$$

$$\text{showev('limit(((2x^2 - 2x + 5)/(3x^2 + x - 6)),x,3))}$$

$$\lim_{x \rightarrow 3} \frac{2x^2 - 2x + 5}{3x^2 + x - 6} = \frac{17}{24}$$

$$\text{plot2d("(2x^2 - 2x + 5)/(3x^2 + x - 6)",-2,10,-10,5):}$$

!images/KALKULUS<sub>L</sub>asigi

4. Fungsi 4

$$f(x) = 4x^2 - 3$$

$$\text{showev('limit((4x^2 - 3),x,0))}$$

$$\lim_{x \rightarrow 0} 4x^2 - 3 = -3$$

$$\text{plot2d("(4x^2 - 3)") :}$$

!images/KALKULUS<sub>L</sub>asigi

5. Fungsi 5

$$f(x) = x^{x^x}$$

$$\text{showev('limit((x^(x(x))),x,0,plus))}$$

$$\lim_{x \downarrow 0} x^{x^x} = 0$$

$$\text{plot2d("(x^(x(x)))",-3,3,-1,7):}$$

!images/KALKULUS<sub>L</sub>asigi

6. Fungsi 6

$$f(x) = 3x \tan x \frac{1}{1 - \cos 4x}$$

$$\text{showev('limit((3xtan(x))/(1-cos(4x)),x,0))}$$

$$3 \left( \lim_{x \rightarrow 0} \frac{x \tan x}{1 - \cos(4x)} \right) = \frac{3}{8}$$

$$\text{plot2d(' (3xtan(x))/(1-cos(4x))', -pi/2, 2pi, 0, 2pi):}$$

!images/KALKULUS\_Lasigi

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).

$$\text{showev('limit(((x+h)^n - x^n)/h, h, 0))//turunan x^n}$$

$$\lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h} = nx^{n-1}$$

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.

Jawab:

Akan ditunjukkan bahwa

$$f'(x) = \lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h} = nx^{n-1}$$

Pertama, ekspansikan  $(x+h)^n$ , yakni:

$$(x+h)^n = \sum_{k=0}^n n k x^{n-k} h^k$$

$$\Leftrightarrow (x+h)^n = n0x^n + n1x^{n-1}h + n2x^{n-2}h^2 + \dots + nnh^n$$

$$\Leftrightarrow (x+h)^n = x^n + nx^{n-1}h + n2x^{n-2}h^2 + n3x^{n-3}h^3 + \dots + h^n$$



Sehingga,  $f'(x)$  menjadi:  $f'(x) = \lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h}$

$$\Leftrightarrow f'(x) = \lim_{h \rightarrow 0} \frac{x^n + nx^{n-1}h + n2x^{n-2}h^2 + n3x^{n-3}h^3 + \dots + h^n - x^n}{h}$$

$$\Leftrightarrow f'(x) = \lim_{h \rightarrow 0} nx^{n-1} + n2x^{n-2}h + n3x^{n-3}h^2 + \dots + h^{n-1}$$

$$\Leftrightarrow f'(x) = nx^{n-1}. \text{ Terbukti.}$$

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.

Jawab:

*Akanditunjukkanbahwa*

$$\lim_{h \rightarrow 0} \frac{\sin(x+h) - \sin x}{h} = \cos x$$

Diketahui bahwa:

$$1). \sin(x+h) = \sin x \cos h + \cos x \sin h$$

$$2). \lim_{h \rightarrow 0} \frac{1 - \cos h}{h} = 0$$

$$3). \lim_{h \rightarrow 0} \frac{\sin h}{h} = 1$$

$$\lim_{h \rightarrow 0} \frac{\sin(x+h) - \sin x}{h}$$

$$= \lim_{h \rightarrow 0} \frac{\sin x \cos h + \cos x \sin h - \sin x}{h}$$

$$= \lim_{h \rightarrow 0} \left[ -\sin x \cdot \frac{1 - \cos h}{h} + \cos x \cdot \frac{\sin h}{h} \right]$$

$$= (-\sin x) \left[ \lim_{h \rightarrow 0} \frac{1 - \cos h}{h} + (\cos x) \lim_{h \rightarrow 0} \frac{\sin h}{h} \right]$$

$$= (-\sin x)(0) + (\cos x)(1) = \cos x. \text{ Terbukti.}$$

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.

Jawab:

Bukti:

*Ambil*

$$f(x) = {}^a \log x.$$

$$\lim_{h \rightarrow 0} \frac{{}^a \log(x+h) - {}^a \log x}{h}$$

$$= \lim_{h \rightarrow 0} \frac{{}^a \log \frac{(x+h)}{x}}{h}$$

$$= \lim_{h \rightarrow 0} \frac{{}^a \log(1 + \frac{h}{x})}{h}$$

$$= \lim_{h \rightarrow 0} \frac{{}^a \log(1 + \frac{h}{x})}{\frac{h}{x} x}$$

$$= \lim_{h \rightarrow 0} \frac{\frac{x}{h} \cdot {}^a \log(1 + \frac{h}{x})}{x}$$

$$= \lim_{h \rightarrow 0} \frac{{}^a \log(1 + \frac{h}{x}) \frac{x}{h}}{x}$$

$$= \frac{\lim_{h \rightarrow 0} {}^a \log(1 + \frac{h}{x}) \frac{x}{h}}{\lim_{h \rightarrow 0} x}$$

$$= \frac{1}{x \cdot e \log a}$$

$$= \frac{1}{x \cdot \ln a}$$

Menggunakan hasil di atas, maka:

$$\frac{d \ln x}{dx} = \frac{d e \log x}{dx} = \frac{1}{x \cdot \ln e} = \frac{1}{x}. \text{ Terbukti.}$$

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))

$$\lim_{h \rightarrow 0} \frac{e^h - 1}{h} = 1$$

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

x x

showev('limit((f(x+h) - f(x))/h,h,0))/turunan f(x) = x^x

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h} = \text{infinity}$$

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))//turunanf(x) = x^x`

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h} = x^x (\log x + 1)$$

`forget(x 0) // jangan lupa, lupakan asumsi untuk kembali ke semula`

`[x gt; 0]`

`forget(x;0)`

`[x lt; 0]`

`facts()`

`[]`

`showev('limit((asin(x+h)-asin(x))/h,h,0))//turunanarcsin(x)`

$$\lim_{h \rightarrow 0} \frac{\arcsin(x+h) - \arcsin x}{h} = \frac{1}{\sqrt{1-x^2}}$$

`showev('limit((tan(x+h)-tan(x))/h,h,0))//turunantan(x)`

$$\lim_{h \rightarrow 0} \frac{\tan(x+h) - \tan x}{h} = \frac{1}{\cos^2 x}$$

`function f(x) = sinh(x) // definisikan f(x)=sinh(x)`

`sinh(x)`

`function df(x) = limit((f(x+h)-f(x))/h,h,0); df(x)//df(x) = f'(x)`

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

Hasilnya adalah  $\cosh(x)$ , karena

$$\frac{e^x + e^{-x}}{2} = \cosh(x).$$

`plot2d(["f(x)", "df(x)"], -pi, pi, color=[blue, red]):`

`![images/KALKULUS_Lasigi`

`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), seperti contoh-contoh tersebut. Gambar grafik fungsi asli dan fungsi turunannya pada sumbu koordinat yang sama.

Jawab:

1. Fungsi 1

`function f(x) := x^2`

*showev('limit(((x+h)^2-x^2)/h),h,0))//turunanx^2*

$$\lim_{h \rightarrow 0} \frac{(x+h)^2 - x^2}{h} = 2x$$

function df(x) = limit(((x+h)^2-x^2)/h,h,0);df(x)// df(x) = f'(x)

$$2x$$

plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]), label("f(x)",2,0.6), label("df(x)",2,0.17):  
![images/KALKULUS\_Lasigi

2. Fungsi 2

function f(x) := sin(x)cos(x)

*showev('limit(((sin(x+h)cos(x+h))-sin(x)cos(x))/h,h,0))//turunansin(x) × cos(x)*

$$\lim_{h \rightarrow 0} \frac{\cos(x+h) \sin(x+h) - \cos x \sin x}{h} = \cos^2 x - \sin^2 x$$

function df(x) = limit(((sin(x+h)cos(x+h))-sin(x)cos(x))/h,h,0); df(x)//df(x) = f'(x)

$$\cos^2 x - \sin^2 x$$

plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]), label("f(x)",1,0), label("df(x)",2.3,1.2):  
![images/KALKULUS\_Lasigi

3. Fungsi 3

function f(x) := sqrt(x)4

*showev('limit((sqrt(x+h)4-sqrt(x)4)/h,h,0))//turunansqrt(x)4*

$$\lim_{h \rightarrow 0} \frac{4\sqrt{x+h} - 4\sqrt{x}}{h} = \frac{2}{\sqrt{x}}$$

function df(x) = limit((sqrt(x+h)4-sqrt(x)4)/h,h,0); df(x)//df(x) = f'(x)

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

plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]), label("f(x)",-2,11), label("df(x)",-2,-10):

![images/KALKULUS\_Lasigi

4. Fungsi 4

function f(x) := cos(1/x)

*showev('limit((cos(1/(x+h))-cos(1/x))/h,h,0))//turunancos(1/x)*

$$\lim_{h \rightarrow 0} \frac{\cos\left(\frac{1}{x+h}\right) - \cos\left(\frac{1}{x}\right)}{h} = \frac{\sin\left(\frac{1}{x}\right)}{x^2}$$

function df(x) = limit((cos(1/(x+h))-cos(1/x))/h,h,0); df(x)//df(x) = f'(x)

$$\frac{\sin\left(\frac{1}{x}\right)}{x^2}$$

plot2d(["f(x)", "df(x)"], -pi, pi, color=[blue, red]), label("f(x)", 2, 0.4), label("df(x)", 1, -0.5):

![images/KALKULUS\_Lasigi

5. Fungsi 5

function f(x) := (log(x))^5

showev('limit(((log(x+h))^5 - (log(x))^5)/h, h, 0))/turunan(log(x))^5

$$\lim_{h \rightarrow 0} \frac{\log^5(x+h) - \log^5 x}{h} = \frac{5 \log^4 x}{x}$$

function df(x) = limit(((log(x+h))^5 - (log(x))^5)/h, h, 0); df(x) // df(x) = f'(x)

$$\frac{5 \log^4 x}{x}$$

plot2d(["f(x)", "df(x)"], -50, 100, -10, 50, color=[blue, red]), label("f(x)", 25, 35), label("df(x)", 50, 1):

![images/KALKULUS\_Lasigi

6. Fungsi 6

function f(x) := sqrt(tan(x))

showev('limit((sqrt(tan(x+h))-sqrt(tan(x)))/h, h, 0))/turunanexp(x) × cos(x)

$$\lim_{h \rightarrow 0} \frac{\sqrt{\tan(x+h)} - \sqrt{\tan x}}{h} = \frac{1}{2 \cos^2 x \sqrt{\tan x}}$$

function df(x) = limit((sqrt(tan(x+h))-sqrt(tan(x)))/h, h, 0); df(x) // df(x) = f'(x)

$$\frac{1}{2 \cos^2 x \sqrt{\tan x}}$$

plot2d(["f(x)", "df(x)"], -10, 10, -10, 10, color=[blue, red]), label("f(x)", 4.5, 0), label("df(x)", 5.5, 5):

![images/KALKULUS\_Lasigi

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)`

$$\int_0^\pi \sin x dx = 2$$

`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(2x+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(2x+1), x, 0, 2)`

$$\int_0^2 x^2 \sqrt{2x+1} dx = \frac{2 \cdot 5^{\frac{5}{2}}}{21} - \frac{2}{105}$$

*ratsimp*(

$$\int_0^2 x^2 \sqrt{2x+1} \, dx = \frac{2 \cdot 5^{\frac{7}{2}} - 2}{105}$$

*showev*('integrate((sin(sqrt(x)+a)E^sqrt(x))/sqrt(x),x,0,pi^2))

$$\int_0^{\pi^2} \frac{\sin(\sqrt{x}+a) e^{\sqrt{x}}}{\sqrt{x}} \, dx = (-e^{\pi} - 1) \sin a + (e^{\pi} + 1) \cos a$$

*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);f(x)

$$e^{-x^2}$$

*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:

$$\int_0^{\pi} e^{-x^2} \, dx$$

x=0:0.1:pi-0.1; plot2d(x,f(x+0.1), bar); plot2d("f(x)",0,pi, add):

![images/KALKULUS<sub>L</sub>asigi

Integral tentu

$$\int_0^{\pi} e^{-x^2} \, dx$$

dapat dihamperi 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:



$$\int_0^{\pi} e^{-x^2} dx = 0.8362196102528469$$

Jumlah tersebut diperoleh dari hasil kali lebar sub-subinterval (=0.1) dan jumlah nilai-nilai  $f(x)$  untuk  $x = 0.1, 0.2, 0.3, \dots, 3.2$ .

```
0.1sum(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,infinity))
```

$$\int_0^{\infty} e^{-x^2} dx = \frac{\sqrt{\pi}}{2}$$

Berikut adalah contoh lain fungsi yang tidak memiliki antiderivatif, sehingga integral tentunya hanya dapat dihitung dengan metode numerik.

```
function f(x) = x^x;f(x)
```

$$x^x$$

```
showev('integrate(f(x),x,0,1))
```

$$\int_0^1 x^x dx = \int_0^1 x^x dx$$

```
x:=0:0.1:1-0.01; plot2d(x,f(x+0.01),bar); plot2d("f(x)",0,1,add):
```

```
!images/KALKULUS_Lasigi
```

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));
```

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 \* diintegrasikan (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.)

Jawab:

1. Fungsi 1

function  $f(x) = 5x^2$ ;  $f(x)$

$$5x^2$$

*showev('integrate(f(x),x))*

$$5 \int x^2 dx = \frac{5x^3}{3}$$

*showev('integrate(f(x),x,2,3))*

$$5 \int_2^3 x^2 dx = \frac{95}{3}$$

$x=0.01:0.03:4$ ; `plot2d(x,f(x+0.01), bar); plot2d("f(x)",2,3, add):`

![images/KALKULUS\_Lasigi

2. Fungsi 2

function  $f(x) = \cos(2x+5)$ ;  $f(x)$


$$\cos(2x + 5)$$

*showev('integrate(f(x),x))*

$$\int \cos(2x + 5) dx = \frac{\sin(2x + 5)}{2}$$

*showev('integrate(f(x),x,pi,2pi))*

$$\int_{\pi}^{2\pi} \cos(2x+5) \, dx = 0$$

x=0:0.05:pi-0.1; plot2d(x,f(x+0.03), bar); plot2d("f(x)",pi,2pi, add):  


3. Fungsi 3

function f(x) = (sin(x))(cos((x)))<sup>2</sup>;f(x)


$$\cos^2 x \sin x$$

*showev('integrate(f(x),x))*

$$\int \cos^2 x \sin x \, dx = -\frac{\cos^3 x}{3}$$

*showev('integrate(f(x),x,0,pi))*

$$\int_0^{\pi} \cos^2 x \sin x \, dx = \frac{2}{3}$$

x=-pi:0.04:pi; plot2d(x,f(x+0.01), bar); plot2d("f(x)",0,pi, add):  


4. Fungsi 4

function f(x) = (x<sup>2</sup>(2-x<sup>3</sup>)<sup>(1/2)</sup>);f(x)


$$x^2 \sqrt{2-x^3}$$

*showev('integrate(f(x),x))*

$$\int x^2 \sqrt{2-x^3} \, dx = -\frac{2(2-x^3)^{\frac{3}{2}}}{9}$$

*showev('integrate(f(x),x,0,1))*

$$\int_0^1 x^2 \sqrt{2-x^3} \, dx = \frac{2^{\frac{5}{2}}}{9} - \frac{2}{9}$$

x=-1:0.04:1; plot2d(x,f(x+0.01), bar); plot2d("f(x)",0,1, add):  


5. Fungsi 5

function f(x) = sqrt(24-x<sup>2</sup>);f(x)

$$\sqrt{24-x^2}$$

*showev('integrate(f(x),x))*

$$\int \sqrt{24-x^2} \, dx = 12 \arcsin\left(\frac{x}{2\sqrt{6}}\right) + \frac{x\sqrt{24-x^2}}{2}$$

*showev('integrate(f(x), x, 1, 2))*

$$\int_1^2 \sqrt{24 - x^2} \, dx = 12 \arcsin\left(\frac{1}{\sqrt{6}}\right) - \frac{24 \arcsin\left(\frac{1}{2\sqrt{6}}\right) + \sqrt{23}}{2} + 2\sqrt{5}$$

*x=-2:0.04:1; plot2d(x,f(x+0.01), bar); plot2d("f(x)",1,2, add):*

*![images/KALKULUS<sub>L</sub>asigi*

*6. Fungsi 6*

*t = makelist(a,a,0,1-0.01,0.01);*

*fx = makelist(f(t[i]+0.01),i,1,length(t));*

*function f(x) = x<sup>2</sup> + 50;f(x)*

$$x^2 + 50$$

*x=0:0.1:pi-0.01; plot2d(x,f(x+0.01), bar); plot2d("f(x)",0,pi, add):*

*![images/KALKULUS<sub>L</sub>asigi*

*0.01sum(f(x+0.01))*

*17.051552*

*7. Fungsi 7*

*t = makelist(a,a,0,1-0.01,0.01);*

*fx = makelist(f(t[i]+0.01),i,1,length(t));*

*function f(x) = cos(x)/x; f(x)*

$$\frac{\cos x}{x}$$

*x=-pi:0.07:pi-0.01; plot2d(x,f(x+0.01), bar); plot2d("f(x)",0,pi, add):*

*![images/KALKULUS<sub>L</sub>asigi*

*0.01sum(f(x+0.01))*

*0.415163991256*

*8. Fungsi 8*

*t = makelist(a,a,0,1-0.01,0.01);*

*fx = makelist(f(t[i]+0.01),i,1,length(t));*

*function f(x) = sqrt(x<sup>2</sup> - 1);f(x)*

$$\sqrt{x^2 - 1}$$

*x=3:0.04:pi-0.01; plot2d(x,f(x+0.01), bar); plot2d("f(x)",0,2, add):*

*![images/KALKULUS<sub>L</sub>asigi*

*0.01sum(f(x+0.01))*

*0.11610107668*

*Luas daerah dibatasi 2 kurva*

*1). Fungsi 1*

*function f(x) = x<sup>3</sup>;f(x)*

$$x^3$$

*function g(x) = x; g(x)*

$$x$$

```
plot2d(["x^4", "x^3"], -2, 2, -1, 2) :
!images/KALKULUS_Lasigi
function h(x) = f(x)-g(x); h(x)
```

$$x^3 - x$$

```
showev('integrate(h(x), x))
```

$$\int x^3 - x \, dx = \frac{x^4}{4} - \frac{x^2}{2}$$

```
solve(f(x) = g(x))
```

$$[x = -1, x = 1, x = 0]$$

```
showev('integrate(h(x), x, 0, 1))//menghitungluasdaerahyangdibatasi2kurva
```

$$\int_0^1 x^3 - x \, dx = -\frac{1}{4}$$

*Arsirandaerahyangdibatasikurva*

```
f(x)dang(x)sebagaiberikut : x = -1 : 0.01 : 1; plot2d(x, f(x), bar, filled, style =
" - ", fillcolor = orange, grid); plot2d(x, g(x), bar, add, filled, style = " -
", fillcolor = white); label("f(x)", 0, 2.1); label("g(x)", 0.5, 0.3) :
!images/KALKULUS_Lasigi
2). Fungsi 2
function f(x) = x^3 + 1; f(x)
```

$$x^3 + 1$$

```
function g(x) = x^2; g(x)
```

$$x^2$$

```
plot2d(["-x^2 + 2", "x^2"], -2, 2, -1, 2) :
!images/KALKULUS_Lasigi
function h(x) = f(x)-g(x); h(x)
```

$$x^3 - x^2 + 1$$

```
solve(f(x) = g(x))
```

$$\left[ x = \frac{\frac{\sqrt{3}i}{2} - \frac{1}{2}}{9 \left( \frac{\sqrt{23}}{2 \cdot 3^{\frac{3}{2}}} - \frac{25}{54} \right)^{\frac{1}{3}}} + \left( \frac{\sqrt{23}}{2 \cdot 3^{\frac{3}{2}}} - \frac{25}{54} \right)^{\frac{1}{3}} \left( -\frac{\sqrt{3}i}{2} - \frac{1}{2} \right) + \frac{1}{3}, x = \left( \frac{\sqrt{23}}{2 \cdot 3^{\frac{3}{2}}} - \frac{25}{54} \right)^{\frac{1}{3}} \left( \frac{\sqrt{3}i}{2} - \frac{1}{2} \right) + \frac{-\frac{\sqrt{3}i}{2} - \frac{1}{2}}{9 \left( \frac{\sqrt{23}}{2 \cdot 3^{\frac{3}{2}}} - \frac{25}{54} \right)^{\frac{1}{3}}} \right]$$

*showev('integrate(h(x), x, -1, 1))//menghitungluasdaerahyangdibatasi2kurva*

$$\int_{-1}^1 x^3 - x^2 + 1 \, dx = \frac{4}{3}$$

x=-1:0.01:1; plot2d(x,f(x), bar, filled,style="-",fillcolor=orange, grid); plot2d(x,g(x), bar, add, filled,style="-",fillcolor=white); label("f(x)",0,2.1); label("g(x)",0.5,0.3):

!images/KALKULUS<sub>L</sub>asigi

Volume benda putar

Menghitung volume hasil perputaran kurva

$$m(x) = x^3 + 1$$

dari x=-1 sampai x=0. Diputar terhadap sumbu-x.

Jawab:

$$\text{function } m(x) = x^4 + 3; m(x)$$

$$x^4 + 3$$

*showev('integrate(pi(m(x))^2, x, -1, 0))//Menghitungvolumehasilperputaranm(x)*

$$\pi \int_{-1}^0 (x^4 + 3)^2 \, dx = \frac{464 \pi}{45}$$

Daerah di bawah kurva yang akan dirotasi terhadap sumbu x sebagai berikut:

plot2d("m(x)",-1,0,-1,2,grid=7, filled, style="/  
"):

!images/KALKULUS<sub>L</sub>asigi

Hasil perputaran m(x) terhadap sumbu x sebagai berikut:

plot3d("m(x)",-1,0,-1,1, rotate,angle=6.3, hue, contour,color=redgreen,height=11):

!images/KALKULUS<sub>L</sub>asigi

Menghitung panjang kurva

Menghitung panjang kurva

$$y = x^2 - x + 1$$

dari x=1 sampai x=3.

$$\text{function } d(x) = x^2 - x + 1; d(x)$$

$$x^2 - x + 1$$

plot2d("d(x)",-5,6): // gambar kurva d(x)

!images/KALKULUS<sub>L</sub>asigi

*showev('limit((d(x+h)-d(x))/h, h, 0))*

$$\lim_{h \rightarrow 0} \frac{(x+h)^2 - x^2 - h}{h} = 2x - 1$$

function dd(x) = limit((d(x+h)-d(x))/h,h,0); dd(x)

$$2x - 1$$

function q(x) = ((dd(x))^2);q(x)

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

showev('integrate(sqrt(1+q(x)),x,1,3))/menghitungpanjangkurva

$$\int_1^3 \sqrt{(2x-1)^2 + 1} dx = \frac{\operatorname{asinh} 5 + 5\sqrt{26}}{4} - \frac{\operatorname{asinh} 1 + \sqrt{2}}{4}$$

Jadi, panjang kurva

$$y=x^2 - x + 1$$

$$dari x = 0 \text{ sampai } x = 4 \text{ adalah}$$

$$S = \operatorname{asinh} 5 + 5\sqrt{26} \frac{\operatorname{asinh}(1) + \sqrt{2}}{4}.$$

### 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 berbagai panduan penggunaan Maxima di software Maxima atau dari Internet.)

Barisan dapat didefinisikan dengan beberapa cara di dalam EMT, di antaranya:

- \* Dengan cara yang sama seperti mendefinisikan vektor dengan \* elemen-elemen beraturan (menggunakan titik dua ":" );
- \* Menggunakan perintah "sequence" dan rumus barisan (suku ke -n);
- \* Menggunakan perintah "iterate" atau "niterate";
- \* Menggunakan fungsi Maxima "create<sub>i</sub>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]

sum(1:2:30), sum(1/(1:2:30))

225 2.33587263431

'sum(k, k, 1, n) = factor(ev(sum(k, k, 1, n), simpsum = true))/simpsum :  
menghitungderetsecarasimbolik

$$\sum_{k=1}^n k = \frac{n(n+1)}{2}$$

'sum(1/(3<sup>k</sup> + k), k, 0, inf) = factor(ev(sum(1/(3<sup>k</sup> + k), k, 0, inf), simpsum = true))

$$\sum_{k=0}^{\infty} \frac{1}{3^k + k} = \sum_{k=0}^{\infty} \frac{1}{3^k + k}$$

Di sini masih gagal, hasilnya tidak dihitung.

'sum(1/x<sup>2</sup>, x, 1, inf) = ev(sum(1/x<sup>2</sup>, x, 1, inf), simpsum = true)/ev : menghitungnilaieksresi

$$\sum_{x=1}^{\infty} \frac{1}{x^2} = \frac{\pi^2}{6}$$

'sum((-1)<sup>(k-1)</sup>/k, k, 1, inf) = factor(ev(sum((-1)<sup>(x-1)</sup>/x, x, 1, inf), simpsum = true))

$$\sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k} = - \sum_{x=1}^{\infty} \frac{(-1)^x}{x}$$

Di sini masih gagal, hasilnya tidak dihitung.

'sum((-1)<sup>k</sup>/(2k-1), k, 1, inf) = factor(ev(sum((-1)<sup>k</sup>/(2k-1), k, 1, inf), simpsum = true))

$$\sum_{k=1}^{\infty} \frac{(-1)^k}{2k-1} = \sum_{k=1}^{\infty} \frac{(-1)^k}{2k-1}$$

ev(sum(1/n!, n, 0, inf), simpsum = true)

$$\sum_{n=0}^{\infty} \frac{1}{n!}$$

Di sini masih gagal, hasilnya tidak dihitung, seharusnya hasilnya e.

assume(abs(x)|1); 'sum(ax<sup>k</sup>, k, 0, inf) = ev(sum(ax<sup>k</sup>, k, 0, inf), simpsum = true), forget(abs(x) < 1);



$$a \sum_{k=0}^{\infty} x^k = \frac{a}{1-x}$$

Deret geometri tak hingga, dengan asumsi rasional antara -1 dan 1.

Deret Taylor

Deret Taylor suatu fungsi  $f$  yang diferensiabel sampai tak hingga di sekitar  $x=a$  adalah:

$$f(x) = \sum_{k=0}^{\infty} \frac{(x-a)^k f^{(k)}(a)}{k!}.$$

' $e^x = \text{taylor}(\exp(x), x, 0, 10) // \text{deret Taylore}^x \text{ disekitar } x = 0, \text{ sampai suku ke } -$   
11

$$e^x = \frac{x^{10}}{3628800} + \frac{x^9}{362880} + \frac{x^8}{40320} + \frac{x^7}{5040} + \frac{x^6}{720} + \frac{x^5}{120} + \frac{x^4}{24} + \frac{x^3}{6} + \frac{x^2}{2} + x + 1$$

' $\log(x) = \text{taylor}(\log(x), x, 1, 10) // \text{deret log}(x) \text{ disekitar } x = 1$

$$\log x = x - \frac{(x-1)^{10}}{10} + \frac{(x-1)^9}{9} - \frac{(x-1)^8}{8} + \frac{(x-1)^7}{7} - \frac{(x-1)^6}{6} + \frac{(x-1)^5}{5} - \frac{(x-1)^4}{4} + \frac{(x-1)^3}{3} - \frac{(x-1)^2}{2} - 1$$

## 6 Geometri

GEOMETRI<sub>L</sub> *LasigiYatindraJago* 23030630008 Nama : *LasigiYatindraJago*

NIM : 23030630008

Kelas : Matematika B

—  
Visualisasi dan Perhitungan Geometri dengan EMT

Euler menyediakan beberapa fungsi untuk melakukan visualisasi dan perhitungan geometri, baik secara numerik maupun analitik (seperti biasanya tentunya, menggunakan Maxima). Fungsi-fungsi untuk visualisasi dan perhitungan 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  $\angle ABC$   
 angleBisector(A, B, C): garis bagi sudut  $\angle ABC$   
 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 membentuk suatu segitiga

doublespread(sa): Spread sudut rangkap Spread  $2*\phi$ , dengan  $sa=\sin(\phi)^2 spreada$ .

Contoh 1: Luas, Lingkaran Luar, Lingkaran Dalam Segitiga

Untuk menggambar objek-objek geometri, langkah pertama adalah menentukan rentang sumbu-sumbu koordinat. Semua objek geometri akan digambar pada satu bidang koordinat, sampai didefinisikan bidang koordinat yang baru.

```
setPlotRange(-0.5,2.5,-0.5,2.5); // mendefinisikan bidang koordinat baru
```

Sekarang tetapkan tiga poin dan plot mereka.

```
A=[1,0]; plotPoint(A,"A"); // definisi dan gambar tiga titik
```

```
B=[0,1]; plotPoint(B,"B");
```

```
C=[2,2]; plotPoint(C,"C");
```

Kemudian tiga segmen.

```
plotSegment(A,B,"c"); // c=AB
```

```
plotSegment(B,C,"a"); // a=BC
```

```
plotSegment(A,C,"b"); // b=AC
```

Fungsi geometri meliputi fungsi untuk membuat garis dan lingkaran. Format garis adalah [a,b,c], yang mewakili garis dengan persamaan  $ax+by=c$ .

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

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

Hitunglah garis tegak lurus yang melalui A pada BC.

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

A

Dan persimpangannya dengan BC.

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

BC

Plot itu.

```
plotPoint(D,value=1); // koordinat D ditampilkan
```

```
aspect(1); plotSegment(A,D): // tampilkan semua gambar hasil plot...()
```

```
!images/GEOMETRI_Lasigi
```

Hitung luas ABC:

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

1.5

Bandingkan dengan rumus determinan.

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

1.5

Cara lain menghitung luas segitiga ABC:

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

1.5

Sudut di C

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

36°52'11.63"

Sekarang lingkaran luar segitiga.

```
c=circleThrough(A,B,C); // lingkaran luar segitiga ABC
```

```
R=getCircleRadius(c); // jari2 lingkaran luar
```

```
O=getCircleCenter(c); // titik pusat lingkaran c
```

```
plotPoint(O,"O"); // gambar titik "O"
```

```
plotCircle(c,"Lingkaran luar segitiga ABC");
```

```

!images/GEOMETRILasigi
Tampilkan koordinat titik pusat dan jari-jari lingkaran luar.
O, R
[1.16667, 1.16667] 1.17851130198
Sekarang akan digambar lingkaran dalam segitiga ABC. Titik pusat lingkaran
dalam adalah titik potong garis-garis bagi sudut.
l=angleBisector(A,C,B); // garis bagi ∠ACB
g=angleBisector(C,A,B); // garis bagi ∠CAB
P=lineIntersection(l,g) // titik potong kedua garis bagi sudut
[0.86038, 0.86038]
Tambahkan semuanya ke plot.
color(5); plotLine(l); plotLine(g); color(1); // gambar kedua garis bagi
sudut
plotPoint(P,"P"); // gambar titik potongnya
r=norm(P-projectToLine(P,lineThrough(A,B))) // jari-jari lingkaran dalam
0.509653732104
plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"); // gam-
bar lingkaran dalam
!images/GEOMETRILasigi
Latihan
1. Tentukan ketiga titik singgung lingkaran dalam dengan sisi-sisi segitiga
ABC.
setPlotRange(-2.5,4.5,-2.5,4.5);
A=[-2,1]; plotPoint(A,"A");
B=[1,-2]; plotPoint(B,"B");
C=[4,4]; plotPoint(C,"C");
2. Gambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut.
plotSegment(A,B,"c")
plotSegment(B,C,"a")
plotSegment(A,C,"b")
aspect(1);
!images/GEOMETRILasigi
3. Tunjukkan bahwa garis bagi sudut yang ke tiga juga melalui titik pusat
lingkaran dalam.
l=angleBisector(A,C,B);
g=angleBisector(C,A,B);
P=lineIntersection(l,g)
[0.581139, 0.581139]
color(5); plotLine(l); plotLine(g); color(1);
plotPoint(P,"P");
r=norm(P-projectToLine(P,lineThrough(A,B)))
1.52896119631
plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC");
!images/GEOMETRILasigi
Jadi, terbukti bahwa garis bagi sudut yang ketiga juga melalui titik pusat
lingkaran dalam.

```

4. Gambar jari-jari lingkaran dalam.

```
r=norm(P-projectToLine(P,lineThrough(A,B)))
```

```
1.52896119631
```

```
plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"):
```

```
![images/GEOMETRI_Lasigi
```

Contoh 2: Geometri Simbolik

Kita dapat menghitung geometri eksak dan simbolik menggunakan Maxima.

File geometri.e menyediakan fungsi yang sama (dan lebih banyak lagi) di Maxima. Namun, kita dapat menggunakan perhitungan simbolis sekarang.

```
A = [1,0]; B = [0,1]; C = [2,2]; // menentukan tiga titik A, B, C
```

Fungsi untuk garis dan lingkaran bekerja seperti fungsi Euler, tetapi memberikan perhitungan simbolis.

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

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

Kita bisa mendapatkan persamaan garis dengan mudah.

```
getLineEquation(c,x,y),solve(
```

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

```
![images/GEOMETRI_Lasigi
```

```
getLineEquation(lineThrough(A,[x1,y1]),x,y)//persamaan garis melalui A dan (x1,y1)
```

$$(x_1 - 1) y - x y_1 = -y_1$$

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

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

```
Q = lineIntersection(c,h) // Q titik potong garis c=BC dan h
```

```
2 6 [-, -] 5 5
```

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

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

```
distance(A,Q)//jarak AQ
```

$$\frac{3}{\sqrt{5}}$$

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

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

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

```
1.178511301977579
```

```
![images/GEOMETRI_Lasigi
```

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

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

*solve(getLineEquation(angleBisector(A,C,B),x,y),y)[1]//persamaangarisbagi < ACB*

$$y = x$$

*P = lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A)); P//titikpotong2garisbagisudut*

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

*P() // hasilnya sama dengan perhitungan sebelumnya*

*[0.86038, 0.86038]*

*Garis dan Lingkaran yang Berpotongan*

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

*A := [1,0]; c=circleWithCenter(A,4);*

*B := [1,2]; C := [2,1]; l=lineThrough(B,C);*

*setPlotRange(5); plotCircle(c); plotLine(l);*

*Perpotongan garis dengan lingkaran menghasilkan dua titik dan jumlah titik potong.*

*P1,P2,f=lineCircleIntersections(l,c);*

*P1, P2,*

*[4.64575, -1.64575] [-0.645751, 3.64575]*

*plotPoint(P1); plotPoint(P2):*

*![images/GEOMETRI<sub>Lasigi</sub>*

*Begitu pula di Maxima.*

*c = circleWithCenter(A,4) // lingkaran dengan pusat A jari-jari 4*

*[1, 0, 4]*

*l = lineThrough(B,C) // garis l melalui B dan C*

*[1, 1, 3]*

*lineCircleIntersections(l,c)|radcan, //titikpotonglingkarancdangarisl*

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

*Akan ditunjukkan bahwa sudut-sudut yang menghadap busur yang sama adalah sama besar.*

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

*degprint(computeAngle(P1,C,P2))*

*69°17'42.68"*

*C=A+normalize([-4,-3])4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);*

*degprint(computeAngle(P1,C,P2))*

*69°17'42.68"*

*insimg;*

!images/GEOMETRI<sub>Lasigi</sub>

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);

!images/GEOMETRI<sub>Lasigi</sub>

Selanjutnya, kami melakukan hal yang sama di Maxima dengan koordinat umum.

A = [a1,a2]; B = [b1,b2];

c1 = circleWithCenter(A,distance(A,B));

c2 = circleWithCenter(B,distance(A,B));

P = circleCircleIntersections(c1,c2); P1 = P[1]; P2 = P[2];

Persamaan untuk persimpangan cukup terlibat. Tetapi kita dapat menyederhanakannya, jika kita memecahkan y.

g = getLineEquation(lineThrough(P1,P2),x,y);

solve(g,y)

$$\left[ y = \frac{-(2b_1 - 2a_1)x + b_2^2 + b_1^2 - a_2^2 - a_1^2}{2b_2 - 2a_2} \right]$$

Ini memang sama dengan tegak lurus tengah, 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)

$$\left[ y = \frac{(b_2 - a_2)x - a_1b_2 + a_2b_1}{b_1 - a_1} \right]$$

Contoh 3: Rumus Heron

Rumus Heron menyatakan bahwa luas segitiga dengan panjang sisi-sisi a, b dan c adalah:

$$L = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{dengans} = (a+b+c)/2,$$

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:

$$x^2 + y^2 = b^2, \quad (x - a)^2 + y^2 = c^2.$$

```
sol = solve([x^2 + y^2 = b^2, (x - a)^2 + y^2 = c^2], [x, y]) :
!images/GEOMETRILasigi
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):
!images/GEOMETRILasigi
sol = solve([x^2 + y^2 = b^2, (x - a)^2 + y^2 = c^2], [x, y]) :
!images/GEOMETRILasigi
Ekstrak solusi y.
ysol = y with sol[2][2]; ysol
Maxima said: part: invalid index of list or matrix. - an error. To debug
this try: debugmode(true);
Error in: ysol amp;= y with sol[2][2]; ysol...
Kami mendapatkan rumus Heron.
function H(a,b,c) = sqrt(factor((ysola/2)^2));'H(a,b,c)=H(a,b,c)
```

$$H(a, b, [1, 0, 4]) = \frac{|a| |ysol|}{2}$$

'Luas =  $H(3, 4, 5)$  //luassegitigadenganpanjangsisi - sisi3, 4, 5

$$Luas = \frac{3 |ysol|}{2}$$

Tentu saja, setiap segitiga persegi panjang adalah kasus yang terkenal.

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

Variable or function ysol not found. Try "trace errors" to inspect local variables after errors. H: useglobal; return abs(a)\*abs(ysol)/2 Error in:  $H(3,4,5)$  //luas segitiga siku-siku dengan panjang sisi 3, 4, 5 ...

Dan juga jelas, bahwa ini adalah segitiga dengan luas maksimal dan dua sisi 3 dan 4.

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

Variable or function ysol not found. Error in expression:  $3*abs(ysol)/2$   
 $y0=f(x[1], args()); adaptiveevalone : s = Try"traceerrors" to inspect local variables after errors. plot2d : dw/n, dw/n^2, dw/n, auto; args());$

Kasus umum juga berfungsi.



```

solve(diff(H(a,b,c)^2,c)=0,c)
Maxima said: diff: second argument must be a variable; found [1,0,4] - an
error. To debug this try: debugmode(true);
Error in: solve(diff(H(a,b,c)^2,c)=0,c)...
Sekarang mari kita cari himpunan semua titik di mana b+c=d untuk beber-
apa konstanta d. Diketahui bahwa ini adalah elips.
s1 = subst(d-c,b,sol[2]); s1
Maxima said: part: invalid index of list or matrix. - an error. To debug
this try: debugmode(true);
Error in: s1 amp;= subst(d-c,b,sol[2]); s1...
Dan buat fungsi ini.
function fx(a,c,d) = rhs(s1[1]); fx(a,c,d), function fy(a,c,d) = rhs(s1[2]);fy(a,c,d)

```

0

!images/GEOMETRI<sub>L</sub>asigi

Sekarang kita bisa menggambar setnya. Sisi b bervariasi dari 1 hingga 4. Diketahui bahwa kita mendapatkan elips.

```
aspect(1); plot2d(fx(3,x,5),fy(3,x,5),xmin=1,xmax=4,square=1):
```

!images/GEOMETRI<sub>L</sub>asigi

Kita dapat memeriksa persamaan umum untuk elips ini, yaitu.

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

di mana (xm,ym) adalah 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])
```

$$\frac{a^2}{d^2}$$

Kita lihat bahwa tinggi dan luas segitiga adalah maksimal untuk x=0. Jadi luas segitiga dengan a+b+c=d maksimal jika segitiga sama sisi. Kami ingin menurunkan ini secara analitis.

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

$$\left[ \frac{a \, y_{sol}^2}{2} = 0, 0 = 0 \right]$$

Kami mendapatkan beberapa minima, yang termasuk dalam segitiga dengan satu sisi 0, dan solusinya a=b=c=d/3.

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

$$[[a = 0, b = \%r_2]]$$

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 :
```

Closing bracket missing. Found: solve([diff(H(a,b,c)<sup>2</sup>, a) = la, diff(H(a,b,c)<sup>2</sup>, b) = la : Brackets : 1(, 1[Symbolicexpressionexpected.Errorin : ...ve([diff(H(a,b,c)<sup>2</sup>, a) = la, diff(H(a,b,c)<sup>2</sup>, b) = la : ...  
diff(H(a,b,c)<sup>2</sup>, c) = la, a + b + c = d], [a, b, c, la])

Function diff needs a function name, an expression, or a collection (with a function name) for fErrorin : diff(H(a,b,c)<sup>2</sup>, c) = la, a+b+c = d], [a, b, c, la])...

Kita bisa membuat plot situasinya

Pertama-tama atur poin di Maxima.

A = at([x,y],sol[2]); A

Maxima said: part: invalid index of list or matrix. - an error. To debug this try: debugmode(true);

Error in: A amp;= at([x,y],sol[2]); A...

B = [0,0]; B, C = [a, 0]; C

[a, 0]

![images/GEOMETRI<sub>L</sub>asigi

Kemudian atur rentang plot, dan plot titik-titiknya.

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

![images/GEOMETRI<sub>L</sub>asigi

Plot segmen.

plotSegment(mxmeval("A"), mxmeval("C")); ... plotSegment(mxmeval("B"), mxmeval("C"));  
... plotSegment(mxmeval("B"), mxmeval("A"));

Floating point error! plotLabel: ctext(text,c+[v[1],-v[2]]\*d/h-[0,textheight()/2]);

Try "trace errors" to inspect local variables after errors. plotSegment: plotLabel(name,A+v/2,[-v[2],v[1]],d);

Hitung tegak lurus tengah di Maxima.

h = middlePerpendicular(A,B); g = middlePerpendicular(B,C);

Dan pusat lingkaran.

U = lineIntersection(h,g);

Maxima said: part: invalid index of list or matrix. 0: lineIntersection(g=[4,0,8],h=[-a,0,-a<sup>2</sup>/2]) - anerror.Todebugthis try : debugmode(true);

Error in: U amp;= lineIntersection(h,g); ...

Kami mendapatkan rumus untuk jari-jari lingkaran.

assume(a 0,b 0,c 0); distance(U, B)|radcan

$$\frac{\sqrt{725 - 13 \cdot 2^{\frac{3}{2}} \sqrt{5} \sqrt{7} \sqrt{11}}}{\sqrt{5}}$$

Mari kita tambahkan ini ke plot.

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

![images/GEOMETRI<sub>L</sub>asigi

Menggunakan geometri, kami memperoleh rumus sederhana

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

untuk radiusnya. Kami dapat memeriksa, apakah ini benar dengan Maxima. Maxima akan memfaktorkan ini hanya jika kita kuadratkan.

$c^2/\sin(\text{computeAngle}(A,B,C))^2|factor$

Maxima said: expt: undefined: 0 to a negative exponent. – an error. To debug this try: debugmode(true);

Error in:  $c^2/\sin(\text{computeAngle}(A,B,C))^2|factor...$

Contoh 4: Garis Euler dan Parabola

Garis Euler adalah garis yang ditentukan dari sembarang segitiga yang tidak sama sisi. Ini adalah garis tengah segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga, termasuk orthocenter, circumcenter, centroid, titik Exeter dan pusat lingkaran sembilan titik segitiga.

Untuk demonstrasi, kami menghitung dan memplot garis Euler dalam sebuah segitiga.

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

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

Untuk memplot objek geometris, kami menyiapkan area plot, dan menambahkan titik ke sana. Semua plot objek geometris ditambahkan ke plot saat ini.

$\text{setPlotRange}(3); \text{plotPoint}(A,"A"); \text{plotPoint}(B,"B"); \text{plotPoint}(C,"C");$

Kita juga bisa menambahkan sisi segitiga.

$\text{plotSegment}(A,B,""); \text{plotSegment}(B,C,""); \text{plotSegment}(C,A,"");$

!images/GEOMETRI<sub>Lasigi</sub>

Berikut adalah luas segitiga, menggunakan rumus determinan. Tentu saja, kita harus mengambil nilai absolut dari hasil ini.

$\text{areaTriangle}(A,B,C)$

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

Kita dapat menghitung koefisien sisi c.

$c = \text{lineThrough}(A,B)$

$[-1, 3, -2]$

Dan juga dapatkan rumus untuk baris ini.

$\text{getLineEquation}(c,x,y)$

$$3y - x = -2$$

Untuk bentuk Hesse, kita perlu menentukan sebuah titik, sehingga titik tersebut berada di sisi positif dari bentuk Hesse. Memasukkan titik menghasilkan jarak positif ke garis.

$\text{getHesseForm}(c,x,y,C), \text{at}(\$

$$\frac{7}{\sqrt{10}}$$

!images/GEOMETRI<sub>Lasigi</sub>

Sekarang kita hitung lingkaran luar ABC.

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

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

```
O = getCircleCenter(LL); O
```

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

Gambarkan lingkaran dan pusatnya. Cu dan U adalah simbolis. Kami mengevaluasi ekspresi ini untuk Euler.

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

```
![images/GEOMETRILasigi
```

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.

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

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

Tambahkan ke plot kami.

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

```
![images/GEOMETRILasigi
```

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
```

```
![images/GEOMETRILasigi
```

Teorinya memberitahu kita MH=2\*MO. Kita perlu menyederhanakan dengan radcan untuk mencapai ini.

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

2

Fungsi termasuk fungsi untuk sudut juga.

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

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

60°15'18.43"

Persamaan untuk pusat incircle tidak terlalu bagus.

Q = lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A))—radcan;

Q

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

Mari kita hitung juga ekspresi untuk jari-jari lingkaran yang tertulis.

r = distance(Q,projectToLine(Q,lineThrough(A,B)))—ratsimp; r

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

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

Mari kita tambahkan ini ke plot.

color(5); plotCircle(LD());

![images/GEOMETRI<sub>L</sub>asigi

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):

![images/GEOMETRI<sub>L</sub>asigi

Ini seharusnya menjadi beberapa fungsi, tetapi pemecah default Maxima hanya dapat menemukan solusinya, jika kita kuadratkan persamaannya. Akibatnya, kami mendapatkan solusi palsu.

akar = solve(getHesseForm(lineThrough(A,B),x,y,C)<sup>2</sup>-distance([x,y],C)<sup>2</sup>,y)

[y = - 3 x - sqrt(70) sqrt(9 - 2 x) + 26, y = - 3 x + sqrt(70) sqrt(9 - 2 x) +

26]

Solusi pertama adalah

maxima: akar[1]

Menambahkan solusi pertama ke plot menunjukkan, bahwa itu memang jalan yang kita cari. Teorinya memberi tahu kita bahwa itu adalah parabola yang diputar.

plot2d(rhs(akar[1]),add=1):

![images/GEOMETRI<sub>L</sub>asigi

function g(x) = rhs(akar[1]); 'g(x) = g(x)//fungsiyangmendefinisikankurvadiatas

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

```
T = [-1, g(-1)]; // ambil sebarang titik pada kurva tersebut
dTc = distance(T,C); fullratsimp(dTC),float(
```

2.135605779339061

```
! [images/GEOMETRI_Lasigi
```

```
U = projectToLine(T,lineThrough(A,B)); U //proyeksi T padagaris 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(
```

2.135605779339061

```
! [images/GEOMETRI_Lasigi
```

Ternyata jarak T ke C sama dengan jarak T ke AB. Coba Anda pilih titik T yang lain dan ulangi perhitungan-perhitungan di atas untuk menunjukkan bahwa hasilnya juga sama.

Contoh 5: Trigonometri Rasional

Ini terinspirasi dari ceramah N.J.Wildberger. Dalam bukunya "Divine Proportions", Wildberger mengusulkan untuk mengganti pengertian klasik tentang jarak dan sudut dengan kuadrat 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 hanya dapat dilakukan dengan kertas dan pensil. 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 hanya mengevaluasi perkiraan numerik.

```
load geometry;
```

Untuk pengenalan pertama, kami menggunakan segitiga persegi panjang dengan proporsi Mesir terkenal 3, 4 dan 5. Perintah berikut adalah perintah Euler untuk merencanakan geometri bidang yang terdapat dalam file Euler "geometry.e".

```
C:=[0,0]; A:=[4,0]; B:=[0,3]; ... setPlotRange(-1,5,-1,5); ... plot-
Point(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ... plotSegment(B,A,"c");
plotSegment(A,C,"b"); plotSegment(C,B,"a"); ... insimg(30);
```

```
! [images/GEOMETRI_Lasigi
```

Tentu saja,

$$\sin(w_a) = \frac{a}{c},$$

di mana  $wa$  adalah sudut di A. Cara yang biasa untuk menghitung sudut ini, adalah dengan mengambil invers dari fungsi sinus. Hasilnya adalah sudut yang tidak dapat dicerna, yang hanya dapat dicetak kira-kira.

```
wa := arcsin(3/5); degprint(wa)
36°52'11.63"
```

Trigonometri rasional mencoba menghindari hal ini.

Gagasan pertama trigonometri rasional adalah kuadran, yang menggantikan jarak. Sebenarnya, itu hanya jarak kuadrat. Berikut ini,  $a$ ,  $b$ , dan  $c$  menunjukkan kuadrat dari sisi-sisinya.

Teorema Pythagoras menjadi  $a+b=c$ .

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

```
25 = 25
```

Pengertian kedua dari trigonometri rasional adalah penyebaran. Spread mengukur pembukaan antar baris. Ini adalah 0, jika garis-garisnya sejajar, dan 1, jika garis-garisnya persegi panjang. Ini adalah kuadrat 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 sembarang segitiga siku-siku dengan salah satu sudut di A.

```
sa = a/c; sa
```

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

Ini lebih mudah dihitung daripada sudut, tentu saja. Tetapi Anda kehilangan properti bahwa sudut dapat ditambahkan dengan mudah.

Tentu saja, kita dapat mengonversi nilai perkiraan untuk sudut  $wa$  menjadi sprad, dan mencetaknya sebagai pecahan.

```
fracprint(sin(wa)^2)
9/25
```

Hukum kosinus trigonometri klasik diterjemahkan menjadi "hukum silang" berikut.

$$(c + b - a)^2 = 4bc(1 - s_a)$$

Di sini  $a$ ,  $b$ , dan  $c$  adalah kuadrat dari sisi-sisi segitiga, dan  $sa$  adalah penyebaran sudut A. Sisi  $a$ , seperti biasa, berhadapan dengan sudut A.

Hukum ini diimplementasikan dalam file `geometri.e` yang kami muat ke Euler.

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

$$\left[ \left( bb - aa + \frac{7}{6} \right)^2, \left( bb - aa + \frac{7}{6} \right)^2, \left( bb - aa + \frac{5}{3\sqrt{2}} \right)^2 \right] = \left[ \frac{14bb(1-saa)}{3}, \frac{14bb(1-saa)}{3}, \frac{5 \cdot 2^{\frac{3}{2}} bb(1-sa)}{3} \right]$$

Dalam kasus kami, kami mendapatkan

$$\text{crosslaw}(a, b, c, sa)$$

$$1024 = 1024$$

Mari kita gunakan crosslaw ini untuk mencari spread di A. Untuk melakukan ini, kita buat crosslaw untuk kuadran a, b, dan c, dan selesaikan untuk spread yang tidak diketahui sa.

Anda dapat melakukannya dengan tangan dengan mudah, tetapi saya menggunakan Maxima. Tentu saja, kami mendapatkan hasilnya, kami sudah memilikinya.

$$\text{crosslaw}(a, b, c, x), \text{solve}(\text{$$

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

!images/GEOMETRI<sub>L</sub>asigi

Kita sudah tahu ini. Definisi spread adalah kasus khusus dari crosslaw.

Kita juga dapat menyelesaikan ini untuk umum a,b,c. Hasilnya adalah rumus yang menghitung penyebaran sudut segitiga yang diberikan kuadrat dari ketiga sisinya.

$$\text{solve}(\text{crosslaw}(aa, bb, cc, x), x)$$

$$\left[ \left[ \frac{168 \, bb \, x + 36 \, bb^2 + (-72 \, aa - 84) \, bb + 36 \, aa^2 - 84 \, aa + 49}{36}, \frac{168 \, bb \, x + 36 \, bb^2 + (-72 \, aa - 84) \, bb + 36 \, aa^2 - 84 \, aa + 49}{36} \right] \right]$$

Kita bisa membuat fungsi dari hasilnya. Fungsi seperti itu sudah didefinisikan dalam file geometri.e dari Euler.

$$\text{spread}(a, b, c)$$

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

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

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

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

$$\text{spread}(a, a, 4a/7)$$

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

Ini adalah sudut dalam derajat.

$$\text{degprint}(\arcsin(\text{sqrt}(6/7)))$$

$$67^{\circ}47'32.44''$$



Contoh lain

Sekarang, mari kita coba contoh yang lebih maju.

Kami mengatur tiga sudut segitiga sebagai berikut.

```
A:=[1,2]; B:=[4,3]; C:=[0,4]; ... setPlotRange(-1,5,1,7); ... plot-
Point(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ... plotSegment(B,A,"c");
plotSegment(A,C,"b"); plotSegment(C,B,"a"); ... insimg;
```

!images/GEOMETRI<sub>L</sub>asigi

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 mengandung fungsi untuk kuadran antara dua titik.

Dalam contoh berikut, karena c+b bukan a, maka segitiga itu bukan persegi panjang.

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

17

!images/GEOMETRI<sub>L</sub>asigi

!images/GEOMETRI<sub>L</sub>asigi

Pertama, mari kita hitung sudut tradisional. Fungsi computeAngle menggunakan metode biasa berdasarkan hasil kali titik dua vektor. Hasilnya adalah beberapa pendekatan floating point.

$$A = \langle 1, 2 \rangle \quad B = \langle 4, 3 \rangle, \quad C = \langle 0, 4 \rangle$$

$$\mathbf{a} = C - B = \langle -4, 1 \rangle, \quad \mathbf{c} = A - B = \langle -3, -1 \rangle, \quad \beta = \angle ABC$$

$$\mathbf{a} \cdot \mathbf{c} = |\mathbf{a}| \cdot |\mathbf{c}| \cos \beta$$

$$\cos \angle ABC = \cos \beta = \frac{\mathbf{a} \cdot \mathbf{c}}{|\mathbf{a}| \cdot |\mathbf{c}|} = \frac{12 - 1}{\sqrt{17} \sqrt{10}} = \frac{11}{\sqrt{17} \sqrt{10}}$$

$wb = computeAngle(A, B, C); wb, (wb/pi180)()$

$$\arccos \left( \frac{11}{\sqrt{10} \sqrt{17}} \right)$$

32.4711922908

Dengan menggunakan pensil dan kertas, kita dapat melakukan hal yang sama dengan hukum silang. Kami memasukkan kuadran a, b, dan c ke dalam hukum silang dan menyelesaikan x.

$crosslaw(a, b, c, x), solve($

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

!images/GEOMETRI<sub>L</sub>asigi

Yaitu, apa yang dilakukan oleh penyebaran fungsi yang didefinisikan dalam "geometry.e".

sb = spread(b,a,c); sb

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

Maxima mendapatkan hasil yang sama menggunakan trigonometri biasa, jika kita memaksanya. Itu menyelesaikan istilah  $\sin(\arccos(\dots))$  menjadi hasil pecahan. Sebagian besar siswa tidak dapat melakukan ini.

$\sin(\text{computeAngle}(A, B, C))^2$

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

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

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

Menurut definisi.

ha = csb; ha

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

Gambar berikut telah dihasilkan dengan program geometri C.a.R., yang dapat menggambar kuadrat dan menyebar.

image: (20) Rational $_{GeometryC.a.R.}.png$

Menurut definisi, panjang ha adalah akar kuadrat dari kuadratnya.

$\sqrt{ha}$

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

Sekarang kita dapat menghitung luas segitiga. Jangan lupa, bahwa kita berhadapan dengan kuadrat!

$\sqrt{ha}\sqrt{a}/2$

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

Rumus determinan biasa menghasilkan hasil yang sama.

$\text{areaTriangle}(B, A, C)$

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

Rumus Bangau

Sekarang, mari kita selesaikan masalah ini secara umum!

remvalue(a,b,c,sb,ha);

Pertama kita hitung spread di B untuk segitiga dengan sisi a, b, dan c. Kemudian kita menghitung luas kuadrat ("quadrea"?), faktorkan dengan Maxima, dan kita mendapatkan rumus Heron yang terkenal.

Memang, ini sulit dilakukan dengan pensil dan kertas.

$\text{spread}(b^2, c^2, a^2), \text{factor}(\text{$

$$\frac{(-c + b + a)(c - b + a)(c + b - a)(c + b + a)}{16}$$

!images/GEOMETRI<sub>L</sub>asigi

Aturan Triple Spread

Kerugian dari spread adalah mereka tidak lagi hanya menambahkan sudut yang sama.

Namun, tiga spread dari sebuah segitiga memenuhi aturan "triple spread" berikut.

$\text{remvalue}(sa, sb, sc); \text{triplespread}(sa, sb, sc)$

$$(sc + sb + sa)^2 = 2(sc^2 + sb^2 + sa^2) + 4sa sb sc$$

Aturan ini berlaku untuk setiap tiga sudut yang menambah 180°.

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

Sejak menyebar

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

sama, aturan triple spread juga benar, jika

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

Karena penyebaran sudut negatif adalah sama, aturan penyebaran rangkap tiga juga berlaku, jika

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

Misalnya, kita dapat menghitung penyebaran sudut 60°. Ini 3/4. Persamaan memiliki solusi kedua, bagaimanapun, di mana semua spread adalah 0.

$\text{solve}(\text{triplespread}(x, x, x), x)$

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

Sebaran 90° jelas 1. Jika dua sudut dijumlahkan menjadi 90°, sebarannya menyelesaikan persamaan sebaran rangkap tiga dengan a,b,1. Dengan perhitungan berikut kita mendapatkan a+b=1.

$\text{triplespread}(x, y, 1), \text{solve}(\text{$

$$[x = 1 - y]$$

!images/GEOMETRI<sub>L</sub>asigi

Karena sebaran  $180^\circ$ -t sama dengan sebaran t, rumus sebaran rangkap tiga juga berlaku, jika satu sudut adalah jumlah atau selisih dua sudut lainnya.

Jadi kita dapat menemukan penyebaran sudut berlipat ganda. Perhatikan bahwa ada dua solusi lagi. Kami membuat ini fungsi.

$\text{solve}(\text{triplespread}(a, a, x), x), \text{functiondoublespread}(a) = \text{factor}(\text{rhs}(\text{$

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

- 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;

!images/GEOMETRI<sub>L</sub>asigi

Mari kita hitung panjang garis bagi sudut di A. Tetapi kita ingin menyelesaikannya untuk umum a,b,c.

remvalue(a,b,c);

Jadi pertama-tama kita hitung penyebaran sudut yang dibagi dua di A, dengan menggunakan rumus sebaran rangkap tiga.

Masalah dengan rumus ini muncul lagi. Ini memiliki dua solusi. Kita harus memilih yang benar. Solusi lainnya mengacu pada sudut terbelah  $180^\circ$ -wa.

$\text{triplespread}(x, x, a/(a+b)), \text{solve}(\text{$

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

!images/GEOMETRI<sub>L</sub>asigi

!images/GEOMETRI<sub>L</sub>asigi

Mari kita periksa persegi panjang Mesir.

$\text{sa2with}[a = 3^2, b = 4^2]$

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

Kami dapat mencetak sudut dalam Euler, setelah mentransfer penyebaran ke radian.

$\text{wa2} := \arcsin(\sqrt{1/10}); \text{degprint}(\text{wa2})$

$18^\circ 26' 5.82''$

Titik P adalah perpotongan garis bagi sudut dengan sumbu y.

$P := [0, \tan(\text{wa2})/4]$

$[0, 1.33333]$

$\text{plotPoint}(P, "P"); \text{plotSegment}(A, P):$

!images/GEOMETRI<sub>L</sub>asigi

Mari kita periksa sudut dalam contoh spesifik kita.

$\text{computeAngle}(C, A, P), \text{computeAngle}(P, A, B)$

$0.321750554397 \quad 0.321750554397$

Sekarang kita hitung panjang garis bagi AP.

Kami menggunakan teorema sinus dalam segitiga APC. Teorema ini menyatakan bahwa

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

berlaku dalam segitiga apa pun. Kuadratkan, itu diterjemahkan ke dalam apa yang disebut "hukum penyebaran"

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

di mana a,b,c menunjukkan qudrances.

Karena spread CPA adalah  $1-sa^2$ , kita dapatkan darinya bisa  $1=b/(1-sa^2)$  dan dapat menghitung bisa (kuadran dari garis-bagi sudut).

`factor(ratsimp(b/(1-sa2))); bisa =`

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

Mari kita periksa rumus ini untuk nilai-nilai Mesir kita.

`sqrt(mxmeval("at(bisa,[a=3^2,b=4^2])")), distance(A,P)`  
 4.21637021356 4.21637021356

Kita juga dapat menghitung P menggunakan rumus spread.

`py=factor(ratsimp(sa2bisa)); py`

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

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"); plot-
Segment(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;
![images/GEOMETRI_Lasigi
```

Kita dapat menggunakan Maxima untuk menyelesaikan rumus penyebaran rangkap tiga untuk sudut-sudut di pusat O untuk r. Jadi kita mendapatkan rumus untuk jari-jari kuadrat dari pericircle dalam hal kuadrat dari sisi.

Kali ini, Maxima menghasilkan beberapa nol kompleks, yang kita abaikan.

`remvalue(a,b,c,r); //` hapus nilai-nilai sebelumnya untuk perhitungan baru  
`rabc = rhs(solve(triplespread(spread(b,r,r),spread(a,r,r),spread(c,r,r)),r)[4]);`

*rabc*

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

Kita dapat menjadikannya sebagai fungsi Euler.

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

Mari kita periksa hasilnya untuk poin A,B,C.

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

Jari-jarinya memang 1.

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

```
1
```

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

```
spread(b, a, c)rabcratsimp
```

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

Sebenarnya spreadnya adalah  $b/(4r)$ , dan kita melihat bahwa sudut chord dari chord b adalah setengah dari sudut pusat.

```
doublespread(b/(4r)) - 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");
```

```
!images/GEOMETRI_Lasigi
```

dan grafiknya juga agak sederhana: bagian atas kerucut:

```
plot3d("d1",xmin=-2,xmax=0,ymin=-2,ymax=0):
```

```
!images/GEOMETRI_Lasigi
```

Tentu saja minimal 0 dicapai di A.

Dua poin

Sekarang kita lihat fungsi MA+MB dimana A dan B adalah dua titik (tetap). Ini adalah "fakta yang diketahui" bahwa kurva level adalah elips, titik fokusnya adalah A dan B; kecuali untuk AB minimum yang konstan pada segmen [AB]:

```
B=[1,-1];
```

```
function d2(x,y):=d1(x,y)+sqrt((x-B[1])^2 + (y - B[2])^2)
```

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

```
!images/GEOMETRI_Lasigi
```

Grafiknya lebih menarik:

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

!images/GEOMETRI<sub>L</sub>asigi

Pembatasan garis (AB) lebih terkenal:

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

!images/GEOMETRI<sub>L</sub>asigi

Tiga poin

Sekarang hal-hal yang kurang sederhana: Ini sedikit kurang terkenal bahwa  $MA+MB+MC$  mencapai minimum pada satu titik pesawat tetapi untuk menentukan itu kurang sederhana:

1) Jika salah satu sudut segitiga ABC lebih dari  $120^\circ$  (katakanlah di A), maka minimum dicapai pada titik ini (misalnya  $AB+AC$ ).

Contoh:

```
C=[-4,1];
```

```
function d3(x,y):=d2(x,y)+sqrt((x-C[1])^2 + (y - C[2])^2)
```

```
plot3d("d3",xmin=-5,xmax=3,ymin=-4,ymax=4);
```

```
insimg;
```

!images/GEOMETRI<sub>L</sub>asigi

```
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;
```

!images/GEOMETRI<sub>L</sub>asigi

2) Tetapi jika semua sudut segitiga ABC kurang dari  $120^\circ$ , minimumnya adalah pada titik F di bagian dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (maka masing-masing  $120^\circ$ ):

```
C=[-0.5,1];
```

```
plot3d("d3",xmin=-2,xmax=2,ymin=-2,ymax=2):
```

!images/GEOMETRI<sub>L</sub>asigi

```
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;
```



!images/GEOMETRI<sub>L</sub>asigi

Merupakan kegiatan yang menarik untuk mewujudkan gambar di atas dengan perangkat lunak geometri; misalnya, saya tahu soft yang ditulis di Jawa yang memiliki instruksi "garis kontur" ...

Semua ini di atas telah ditemukan oleh seorang hakim Perancis bernama Pierre de Fermat; dia menulis surat kepada dilettants lain seperti pendeta Marin Mersenne dan Blaise Pascal yang bekerja di pajak penghasilan. Jadi titik unik F sedemikian rupa sehingga  $FA+FB+FC$  minimal, disebut titik Fermat segitiga. Tetapi tampaknya beberapa tahun sebelumnya, Torricelli Italia telah menemukan titik ini sebelum Fermat melakukannya! Bagaimanapun tradisinya adalah mencatat poin ini F...

Empat poin

Langkah selanjutnya adalah menambahkan 4 titik D dan mencoba meminimalkan MA+MB+MC+MD; katakan 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 panjang kabel sesedikit mungkin!



```
D=[1,1];
function d4(x,y):=d3(x,y)+sqrt((x-D[1])^2 + (y - D[2])^2)
plot3d("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5):
!
fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
P=(A_B_C_D)'; plot2d(P[1],P[2],points=1,add=1,color=12);
insimg;
!
```

Masih ada minimum dan tidak tercapai di salah satu 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 adalah rasional atau mendekati rasional...

Sekarang ABCD adalah persegi, kami berharap bahwa titik optimal akan menjadi pusat ABCD:

```
C=[-1,1];
plot3d("d4",xmin=-1,xmax=1,ymin=-1,ymax=1):
!
fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
P=(A_B_C_D)'; plot2d(P[1],P[2],add=1,color=12,points=1);
insimg;
!
```

Contoh 7: Bola Dandelin dengan Povray

Anda dapat menjalankan demonstrasi ini, jika Anda telah menginstal Povray, dan pvcengine.exe di jalur program.

Pertama kita hitung 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 geometri.e dari Euler untuk ini.

```
load geometry;
```

Pertama dua garis yang membentuk kerucut.

```
g1 = lineThrough([0,0],[1,a])
[- a, 1, 0]
g2 = lineThrough([0,0],[-1,a])
[- a, - 1, 0]
```

Kemudian saya baris ketiga.

```
g = lineThrough([-1,0],[1,1])
[- 1, 2, 1]
```



Kami merencanakan semuanya sejauh ini.

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

![images/GEOMETRI<sub>L</sub>asigi

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 yang jaraknya sama.

sol = solve(d1<sup>2</sup> = d<sup>2</sup>, 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;

![images/GEOMETRI<sub>L</sub>asigi

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:\Files-Ray\3.7.exe

Kami 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 bidang terbatas pada kerucut.

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

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

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

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

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

```
function turnz(v) := return [-v[2],v[1],v[3]]
```

```
P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]);
```

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

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

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

Kemudian kami menghasilkan dua titik di mana bola menyentuh bidang. Ini adalah fokus dari elips.

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

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

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

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

Selanjutnya kita hitung perpotongan P1P2 dengan bidang.

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

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

Kami menghubungkan titik-titik dengan segmen garis.

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

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

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

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

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

```
pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2],1);
```

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

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

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

Mulai program Povray.

```
povend();
```

```
![images/GEOMETRILasigi
```

Untuk mendapatkan Anaglyph ini kita perlu memasukkan semuanya ke dalam fungsi scene. Fungsi ini akan digunakan dua kali kemudian.

```
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)));
    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 ;/prej. Anda membutuhkan kacamata merah/sian untuk menghar-
gai efek berikut.
```

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

```
!images/GEOMETRILasigi
```

Contoh 8: Geometri Bumi

Dalam buku catatan ini, kami ingin melakukan beberapa perhitungan sferis. Fungsi-fungsi tersebut terdapat dalam file "spherical.e" di folder contoh. Kita perlu memuat file itu terlebih dahulu.

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

Untuk memasukkan posisi geografis, kami menggunakan vektor dengan dua koordinat dalam radian (utara dan timur, nilai negatif untuk selatan dan barat). Berikut 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 spherical).

```
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 ke yang lain pada bola ideal. Vektor ini [pos,jarak] dalam radian. Untuk menghitung jarak di bumi, kita kalikan dengan jari-jari bumi pada garis lintang 7°.

```
br=svector(FMIPA,Solo); degprint(br[1]), br[2]rearth(7°)- km // perkiraan
jarak FMIPA-Solo
65°20'26.60" 53.8945384608
```

Ini adalah perkiraan yang baik. Rutinitas berikut menggunakan perkiraan yang lebih baik. Pada jarak yang begitu pendek hasilnya hampir sama.

```
esdist(FMIPA,Semarang)- " km", // perkiraan jarak FMIPA-Semarang
```

88.0114026318 km

Ada fungsi untuk heading, dengan mempertimbangkan bentuk elips bumi. Sekali lagi, kami mencetak dengan cara yang canggih.

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

65.34°

Sudut segitiga melebihi 180° pada bola.

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

```
degprint(asum)
```

180°0'10.77"

Ini dapat digunakan untuk menghitung luas segitiga. Catatan: Untuk segitiga kecil, ini tidak akurat karena kesalahan pengurangan dalam asum-pi.

```
(asum-pi)rearth(48°)2 - "km2", //perkiraanluassegitigaFMIPA - Solo - Semarang
```

2116.02948749 km<sup>2</sup>

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)- "km2", //perkiraanyangsamadenganfungsiesarea()
```

2123.64310526 km<sup>2</sup>

Kita juga dapat menambahkan vektor ke posisi. Sebuah vektor berisi heading dan jarak, keduanya dalam radian. Untuk mendapatkan vektor, kami menggunakan vektor. Untuk menambahkan vektor ke posisi, kami menggunakan vektor sadd.

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

S 7°34.333' E 110°49.683' S 7°34.333' E 110°49.683'

Fungsi-fungsi ini mengasumsikan bola yang ideal. Hal yang sama di bumi.

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

S 7°34.333' E 110°49.683' S 7°34.333' E 110°49.683'

Mari kita beralih ke contoh yang lebih besar, Tugu Jogja dan Monas Jakarta (menggunakan Google Earth untuk mencari koordinatnya).

```
Tugu=[-7.7833°,110.3661°]; Monas=[-6.175°,106.811944°];
```

```
sposprint(Tugu), sposprint(Monas)
```

S 7°46.998' E 110°21.966' S 6°10.500' E 106°48.717'

Menurut Google Earth, jaraknya adalah 429,66 km. Kami mendapatkan pendekatan yang baik.

```
esdist(Tugu,Monas)- "km", // perkiraan jarak Tugu Jogja - Monas Jakarta
```

431.565659488 km

Judulnya sama dengan judul yang dihitung di Google Earth.

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

294°17'2.85"

Namun, kita tidak lagi mendapatkan posisi target yang tepat, jika kita menambahkan heading dan jarak ke posisi 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'
```

Tentu kita tidak bisa berlayar dengan tujuan yang sama dari satu tujuan ke tujuan lainnya, jika kita ingin menempuh jalur terpendek. Bayangkan, Anda terbang NE mulai dari titik mana pun di bumi. Kemudian Anda akan berputar ke kutub utara. Lingkaran besar tidak mengikuti heading yang konstan!

Perhitungan berikut menunjukkan bahwa kami jauh dari tujuan yang benar, jika kami menggunakan pos yang sama selama perjalanan kami.

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

Sekarang kita tambahkan 10 kali sepersepuluh dari jarak, menggunakan pos ke Monas, kita sampai di Tugu.

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

Hasilnya jauh.

```
sposprint(p), skmprint(esdist(p,Monas))
```

```
S 6°11.250' E 106°48.372' 1.529km
```

Sebagai contoh lain, mari kita ambil dua titik di bumi pada garis lintang yang sama.

```
P1=[30°,10°]; P2=[30°,50°];
```

Jalur terpendek dari P1 ke P2 bukanlah lingkaran garis lintang 30°, melainkan jalur terpendek yang dimulai 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 kita di sepanjang jalan. Untuk tujuan kasar, kami menyesuaikannya pada 1/10 dari total jarak.

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

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

Jaraknya tidak tepat, karena kita akan menambahkan sedikit kesalahan, jika kita mengikuti heading yang sama terlalu lama.

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

```
0.203km
```

Kami mendapatkan perkiraan yang baik, jika kami menyesuaikan pos 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 memplot bumi, dua posisi, dan posisi di antaranya.

```
function testplot ...
useglobal; plotearth; plotpos(Tugu,"Tugu Jogja"); plotpos(Monas,"Tugu Monas");
plotposline(v); endfunction i/prej. Sekarang rencanakan semuanya.
```

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

```
!images/GEOMETRILasigi
```

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

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

```
!images/GEOMETRILasigi
```

MENCOBA RUMUS-RUMUS PADA MATEI DI ATAS

Geometri Simbolik

```
A := [2,0]; B := [0,2]; C := [3,3]; // menentukan tiga titik A, B, C
```

```
c := lineThrough(B,C) // c=BC
```

```
[-1, 3, 6]
```

```
getLineEquation(c,x,y),solve(
```

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

```
!images/GEOMETRILasigi
```

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

```
[3, 1, 6]
```

```
Q = lineIntersection(c,h) // Q titik potong garis c=BC dan h
```

```
6 12 [-, -] 5 5
```

```
projectToLine(A,lineThrough(B,C))//proyeksiApadaBC
```

$$\left[ \frac{6}{5}, \frac{12}{5} \right]$$

```
distance(A,Q)//jarakAQ
```

$$\frac{2^{\frac{5}{2}}}{\sqrt{5}}$$

```
cc = circleThrough(A,B,C); cc//(titikpusatdanjari-jari)lingkaranmelaluiA,B,C
```

$$\left[ \frac{7}{4}, \frac{7}{4}, \frac{5}{2^{\frac{3}{2}}} \right]$$

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

```
1.767766952966368
```

```
!images/GEOMETRILasigi
```

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

$$\arccos\left(\frac{3}{5}\right)$$

*solve(getLineEquation(angleBisector(A,C,B),x,y),y)[1]//persamaangarisbagi < ACB*

$$y = x$$

*P = lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A)); P//titikpotong2*

$$\left[ \frac{\sqrt{2}\sqrt{10}+2}{4}, \frac{\sqrt{2}\sqrt{10}+2}{4} \right]$$

Garis dan Lingkaran yang berpotongan

*A := [2,0]; c = circleWithCenter(A,4);*

*B := [2,3]; C := [3,2]; l = lineThrough(B,C);*

*setPlotRange(5); plotCircle(c); plotLine(l);*

*P1,P2,f=lineCircleIntersections(l,c);*

*P1, P2,*

*[5.89792, -0.897916] [1.10208, 3.89792]*

*plotPoint(P1); plotPoint(P2):*

*![images/GEOMETRI<sub>L</sub>asigi*

*c = circleWithCenter(A,4) // lingkaran dengan pusat A jari-jari 4*

*[2, 0, 4]*

*l = lineThrough(B,C) // garis l melalui B dan C*

*[1, 1, 5]*

*lineCircleIntersections(l,c)|radcan, //titikpotonglingkarancdangarisl*

$$\left[ \left[ \frac{\sqrt{23}+7}{2}, \frac{3-\sqrt{23}}{2} \right], \left[ \frac{7-\sqrt{23}}{2}, \frac{\sqrt{23}+3}{2} \right] \right]$$

*C=A+normalize([-3,-4])4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);*

*degprint(computeAngle(P1,C,P2))*

*57°58'20.06"*

*C=A+normalize([-4,-5])4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);*

*degprint(computeAngle(P1,C,P2))*

*57°58'20.06"*

*insimg;*

*![images/GEOMETRI<sub>L</sub>asigi*

Garis Sumbu

*A :=[3,3]; B :=[-2,-3];*

*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):
!images/GEOMETRILasigi
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];
g = getLineEquation(lineThrough(P1,P2),x,y);
solve(g,y)

```

$$\left[ y = \frac{-(2b_1 - 2a_1)x + b_2^2 + b_1^2 - a_2^2 - a_1^2}{2b_2 - 2a_2} \right]$$

```

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)

```

$$\left[ y = \frac{(b_2 - a_2)x - a_1b_2 + a_2b_1}{b_1 - a_1} \right]$$

Garis Euler dan Parabola

```

A :=[-1.5,-1.5]; B :=[3,0]; C :=[1.5,3];
setPlotRange(3); plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C");
plotSegment(A,B,""); plotSegment(B,C,""); plotSegment(C,A,"");
!images/GEOMETRILasigi
areaTriangle(A,B,C)

```

$$-\frac{63}{8}$$

```

c = lineThrough(A,B)
3 9 9 [-, -, -] 2 2 2
getLineEquation(c,x,y)

```

$$\frac{9y}{2} - \frac{3x}{2} = -\frac{9}{2}$$

```

getHesseForm(c,x,y,C),at(

```

$$\frac{21}{2^{\frac{3}{2}}\sqrt{5}}$$

```

!images/GEOMETRILasigi
LL = circleThrough(A,B,C); getCircleEquation(LL,x,y)

```

$$\left(y - \frac{15}{28}\right)^2 + \left(x - \frac{9}{28}\right)^2 = \frac{2925}{392}$$



O = getCircleCenter(LL); O

$$\left[\frac{9}{28}, \frac{15}{28}\right]$$

plotCircle(LL()); plotPoint(O(),"O");

!images/GEOMETRI<sub>Lasigi</sub>

H = lineIntersection(perpendicular(A,lineThrough(C,B)),... perpendicular(B,lineThrough(A,C))); H

$$\left[\frac{33}{14}, \frac{3}{7}\right]$$

el = lineThrough(H,O); getLineEquation(el,x,y)

$$-\frac{57y}{28} - \frac{3x}{28} = -\frac{9}{8}$$

plotPoint(H(),"H"); plotLine(el(),"Garis Euler");

!images/GEOMETRI<sub>Lasigi</sub>

M = (A+B+C)/3; getLineEquation(el,x,y)with[x = M[1],y = M[2]]

$$-\frac{9}{8} = -\frac{9}{8}$$

plotPoint(M(),"M"); // titik berat

!images/GEOMETRI<sub>Lasigi</sub>

distance(M,H)/distance(M,O)|radcan

2

computeAngle(A,C,B),degprint(

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

60°15'18.43"

Q = lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A))—ratsimp;

Q

$$\left[\frac{\left(3 \cdot 2^{\frac{3}{2}} + 3\right) \sqrt{5} \sqrt{13} - 45 \sqrt{2} + 9}{28}, \frac{(3 \sqrt{2} - 9) \sqrt{5} \sqrt{13} + 15 \cdot 2^{\frac{3}{2}} + 15}{28}\right]$$

r = distance(Q,projectToLine(Q,lineThrough(A,B)))—ratsimp; r

$$\frac{\sqrt{(-369 \sqrt{2} - 279) \sqrt{5} \sqrt{13} + 1035 \sqrt{2} + 5526}}{7 \cdot 2^{\frac{3}{2}}}$$

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

```

color(5); plotCircle(LD()):
![images/GEOMETRI_Lasigi
Contoh lain dari materi trigonometri rasional
A :=[2,3]; B :=[5,4]; C :=[0,5]; ... setPlotRange(-1,5,1,7); ... plot-
Point(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ... plotSegment(B,A,"c");
plotSegment(A,C,"b"); plotSegment(C,B,"a"); ... insimg;
![images/GEOMETRI_Lasigi
distance(A,B)

```

$$\sqrt{10}$$

$$c = \text{quad}(A,B); c, b = \text{quad}(A,C); b, a = \text{quad}(B,C); a,$$

$$26$$

```

![images/GEOMETRI_Lasigi
![images/GEOMETRI_Lasigi
wb = computeAngle(A,B,C); wb,(wb/pi180)()

```

$$\arccos\left(\frac{14}{\sqrt{10}\sqrt{26}}\right)$$

$$29.7448812969$$

$$\text{crosslaw}(a,b,c,x),\text{solve}($$

$$\left[x = \frac{4}{5}\right]$$

```

![images/GEOMETRI_Lasigi
sb = spread(b,a,c); sb

```

$$\frac{16}{65}$$

$$\sin(\text{computeAngle}(A,B,C))^2$$

$$\frac{16}{65}$$

$$ha = \text{csb}; ha$$

$$\frac{32}{13}$$

$$\text{sqrt}(ha)$$

$$\frac{2^{\frac{5}{2}}}{\sqrt{13}}$$

$$\text{sqrt}(ha)\text{sqrt}(a)/2$$

$$\frac{2^{\frac{3}{2}}\sqrt{26}}{\sqrt{13}}$$

*areaTriangle(B, A, C)*

4

Aturan penyebaran 3 kali lipat  
 setPlotRange(1); ... 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"); plot-  
 Segment(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;  
 ![images/GEOMETRI<sub>L</sub>asigi  
 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*

$$-\frac{abc}{c^2 - 2bc + a(-2c - 2b) + b^2 + a^2}$$

function periradius(a,b,c) = rabc;  
 a:=quadrance(B,C); b:=quadrance(A,C); c:=quadrance(A,B);  
 periradius(a,b,c)  
 1  
*spread(b, a, c)rabc|ratsimp*

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

*doublespread(b/(4r)) - 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=[-2,-2];  
 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");

![images/GEOMETRI<sub>L</sub>asigi

dan grafiknya juga agak sederhana: bagian atas kerucut:

plot3d("d1",xmin=-2,xmax=0,ymin=-2,ymax=0):

![images/GEOMETRI<sub>L</sub>asigi

Ternyata setelah mencoba yang bisa hanya dengan memasukkan angka 1, karena ketika memakai angka 2, plot tidak membentuk kerucut diatas.

Dua poin

B=[2,-2];

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):
!images/GEOMETRILasigi
Grafiknya lebih menarik:
plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
!images/GEOMETRILasigi
Pembatasan garis (AB) lebih terkenal:
plot2d("abs(x+1)+abs(x-1)",xmin=-3,xmax=3):
!images/GEOMETRILasigi
Tiga poin
Contoh:
C=[-3,2];
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;
!images/GEOMETRILasigi
fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The mini-
mum is on A");
P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);
insimg;
!images/GEOMETRILasigi
Tetapi jika semua sudut segitiga ABC kurang dari 120 °, minimumnya adalah
pada titik F di bagian dalam segitiga, yang merupakan satu-satunya titik yang
melihat sisi-sisi ABC dengan sudut yang sama (maka masing-masing 120 °):
C=[-1,2];
plot3d("d3",xmin=-2,xmax=2,ymin=-2,ymax=2):
!images/GEOMETRILasigi
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;
!images/GEOMETRILasigi
Empat poin
Langkah selanjutnya adalah menambahkan 4 titik D dan mencoba memi-
nimalkan MA+MB+MC+MD; katakan bahwa Anda adalah operator TV ka-
bel dan ingin mencari di bidang mana Anda harus meletakkan antena sehingga
Anda dapat memberi makan empat desa dan menggunakan panjang kabel sesedikit
mungkin!
D=[2,21];
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):
!images/GEOMETRILasigi
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;
!images/GEOMETRILasigi
Contoh 7: Bola Dandelin dengan Povray

```

```

load geometry;
Pertama dua garis yang membentuk kerucut.
g1 = lineThrough([0,0],[2,a])
[- a, 2, 0]
g2 = lineThrough([0,0],[-2,a])
[- a, - 2, 0]
g = lineThrough([-2,0],[2,2])
[- 2, 4, 4]
setPlotRange(-2,2,0,3);
color(black); plotLine(g(),"")
a:=2; color(blue); plotLine(g1(),""), plotLine(g2(),""):
![images/GEOMETRILasigi
Sekarang kita ambil titik umum pada sumbu y.
P = [0,u]
[0, u]
Hitung jarak ke g1.
d1 = distance(P,projectToLine(P,g1)); d1

```

$$\sqrt{\left(\frac{a^2 u}{a^2 + 4} - u\right)^2 + \frac{4 a^2 u^2}{(a^2 + 4)^2}}$$

Hitung jarak ke g.

```
d = distance(P,projectToLine(P,g)); d
```

$$\sqrt{\left(\frac{u + 4}{5} - u\right)^2 + \frac{(2 u - 2)^2}{25}}$$

Dan temukan pusat kedua lingkaran yang jaraknya sama.

```
sol = solve(d1^2 = d^2, u);sol
```

$$\left[ u = \frac{-\sqrt{5} \sqrt{a^2 + 4} + a^2 + 4}{a^2 - 1}, u = \frac{\sqrt{5} \sqrt{a^2 + 4} + a^2 + 4}{a^2 - 1} \right]$$

Ada dua solusi.

```

u := sol()
[0.558482, 4.77485]
dd := d()
[0.394906, 3.37633]

```

Plot lingkaran ke dalam gambar.

```

color(red);
plotCircle(circleWithCenter([0,u[1]],dd[1]), "");
plotCircle(circleWithCenter([0,u[2]],dd[2]), "");
insimg;
![images/GEOMETRILasigi

```

Latihan

1. Gambarkanlah 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  $\frac{360}{n}$ .

\* Titik-titik sudut segi-n merupakan perpotongan lingkaran luar segi-n \* dan garis-garis yang melalui pusat dan saling membentuk sudut sebesar \* kelipatan  $\frac{360}{n}$ .

\* 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.

Penyelesaian :

load geometry

Numerical and symbolic geometry.

setPlotRange(-3.5,3.5,-3.5,3.5);

A=[-2,-2]; plotPoint(A,"A");

B=[2,-2]; plotPoint(B,"B");

C=[0,3]; plotPoint(C,"C");

plotSegment(A,B,"c");

plotSegment(B,C,"a");

plotSegment(A,C,"b");

aspect(1):

![images/GEOMETRI<sub>L</sub>asigi

c=circleThrough(A,B,C);

R=getCircleRadius(c);

O=getCircleCenter(c);

plotPoint(O,"O");

l=angleBisector(A,C,B);

color(2); plotLine(l); color(1);

plotCircle(c,"Lingkaran luar segitiga ABC");

![images/GEOMETRI<sub>L</sub>asigi

2. Gambarkan 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.

Penyelesaian :

load geometry;

setPlotRange(5); P=[2,0]; Q=[4,0]; R=[0,-4];

plotPoint(P,"P"); plotPoint(Q,"Q"); plotPoint(R,"R");

![images/GEOMETRI<sub>L</sub>asigi

sol = solve([a+b=-c,16a+4b=-c,c=-4],[a,b,c])

[[a = - 1, b = 5, c = - 4]]

Sehingga didapatkan nilai a = -1, b = 5 dan c = -4

function y=-x<sup>2</sup> + 5x - 4

2 - x + 5 x - 4

plot2d("-x<sup>2</sup> + 5x - 4", -5, 5, -5, 5) :

![images/GEOMETRI<sub>L</sub>asigi

3. Gambarkanlah 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.

Penyelesaian :

load geometry

Numerical and symbolic geometry.

setPlotRange(-4.5,4.5,-4.5,4.5);

A=[-3,-3]; plotPoint(A,"A");

B=[3,-3]; plotPoint(B,"B");

C=[3,3]; plotPoint(C,"C");

D=[-3,3]; plotPoint(D,"D");

plotSegment(A,B,"");

plotSegment(B,C,"");

plotSegment(C,D,"");

plotSegment(A,D,"");

aspect(1):

![images/GEOMETRI<sub>Lasigi</sub>

l=angleBisector(A,B,C);

m=angleBisector(B,C,D);

P=lineIntersection(l,m);

color(5); plotLine(l); plotLine(m); color(1);

plotPoint(P,"P");

![images/GEOMETRI<sub>Lasigi</sub>

Dari gambar diatas terlihat bahwa keempat garis bagi sudutnya bertemu di satu titik yaitu titik P.

r=norm(P-projectToLine(P,lineThrough(A,B)));

plotCircle(circleWithCenter(P,r),"Lingkaran dalam segiempat ABCD");

![images/GEOMETRI<sub>Lasigi</sub>

Dari gambar diatas, terlihat bahwa sisi-sisinya merupakan garis singgung lingkaran yang sama yaitu lingkaran dalam segiempat.

Akan ditunjukkan bahwa hasil kali panjang sisi-sisi yang berhadapan sama.

AB=norm(A-B) //panjang sisi AB

6

CD=norm(C-D) //panjang sisi CD

6

AD=norm(A-D) //panjang sisi AD

6

BC=norm(B-C) //panjang sisi BC

6

AB.CD

36

AD.BC

36

Terbukti bahwa hasil kali panjang sisi-sisi yang berhadapan sama yaitu 36. Jadi dapat dipastikan bahwa segiempat tersebut merupakan segiempat garis singgung.

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

Penyelesaian :

Diketahui kedua titik fokus P = [-1,-1] dan Q = [1,-1]

P=[-1,-1]; Q=[1,-1];

function d1(x,y):=sqrt((x-P[1])<sup>2</sup> + (y - P[2])<sup>2</sup>)

Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])<sup>2</sup> + (y - P[2])<sup>2</sup>) + sqrt((x - Q[1])<sup>2</sup> + (y - Q[2])<sup>2</sup>)

fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):

!images/GEOMETRI<sub>Lasigi</sub>

Grafik yang lebih menarik

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

!images/GEOMETRI<sub>Lasigi</sub>

Batasan ke garis PQ

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

!images/GEOMETRI<sub>Lasigi</sub>

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

Penyelesaian :

P=[-1,-1]; Q=[1,-1];

function d1(x,y):=sqrt((x-p[1])<sup>2</sup> + (y - p[2])<sup>2</sup>)

Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])<sup>2</sup> + (y - P[2])<sup>2</sup>) + sqrt((x + Q[1])<sup>2</sup> + (y + Q[2])<sup>2</sup>)

fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):

!images/GEOMETRI<sub>Lasigi</sub>

Grafik yang lebih menarik

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

!images/GEOMETRI<sub>Lasigi</sub>

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

!images/GEOMETRI<sub>Lasigi</sub>

## 7 Statistika

STATISTIKA<sub>LasigiYatindraJago</sub>23030630008Nama : LasigiYatindraJago

NIM : 23030630008



Kelas: Matematika B

EMT untuk Statistika

Di buku catatan ini, kami mendemonstrasikan plot statistik utama, pengujian, dan distribusi di Euler.

Mari kita mulai dengan beberapa statistik deskriptif. Ini bukan pengantar statistik. Jadi, Anda mungkin memerlukan latar belakang untuk memahami detailnya.

Asumsikan pengukuran berikut. Kami ingin menghitung nilai rata-rata dan deviasi standar yang diukur.

```
M=[1000,1004,998,997,1002,1001,998,1004,998,997]; ... median(M), mean(M),  
dev(M),
```

```
999 999.9 2.72641400622
```

Kita dapat memplot plot kotak-dan-kumis untuk datanya. Dalam kasus kami, tidak ada outlier.

```
aspect(1.75); boxplot(M):
```

aspect (1.75) digunakan untuk mengatur rasio aspek dari plot (perbandingan antara lebar dan tinggi).

boxplot(M) digunakan untuk membuat boxplot atau diagram kotak dari data di dalam variabel M. Boxplot adalah visualisasi statistik yang menunjukkan persebaran data, termasuk nilai minimum, median, dan nilai maksimum.

Contoh, kita asumsikan jumlah pria berikut dalam rentang ukuran tertentu.

```
r=155.5:4:187.5; v=[22,71,136,169,139,71,32,8];
```

Berikut adalah alur pendistribusiannya.

```
plot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style=" /"):
```

Kita bisa memasukkan data mentah tersebut ke dalam tabel.

Tabel adalah metode untuk menyimpan data statistik. Tabel kita harus berisi tiga kolom: Awal jangkauan, akhir jangkauan, jumlah pria dalam jangkauan.

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 memerlukan nilai rata-rata dan statistik ukuran lainnya, kita perlu menghitung titik tengah rentang tersebut. Kita bisa menggunakan dua kolom pertama tabel kita untuk ini.

Sumbol "—" digunakan untuk memisahkan kolom, fungsi "writetable" digunakan untuk menulis tabel, dengan opsi "labc" untuk menentukan header kolom.

```
(T[,1]+T[,2])/2
```

```
157.5 161.5 165.5 169.5 173.5 177.5 181.5 185.5
```

```
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 mean dan deviasi sampel dengan frekuensi tertentu.

```
m,d=meandev(M,v); m, d,
```

```
169.901234568 5.98912964449
```

Mari kita tambahkan distribusi nilai normal ke diagram batang di atas. Rumus distribusi normal dengan mean  $m$  dan simpangan baku  $d$  adalah:

Karena nilainya antara 0 dan 1, maka untuk memplotnya pada bar plot 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):
Tabel
```

Di direktori buku catatan ini Anda menemukan file dengan tabel. Data tersebut merupakan hasil survei. Berikut adalah empat baris pertama file tersebut. Datanya berasal dari buku online 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). Kami ingin membaca tabel dari file. Pertama, kami menggunakan terjemahan kami sendiri untuk tokennya.

Untuk ini, kami mendefinisikan kumpulan token. Fungsi `strtokens()` mendapatkan vektor string token dari string tertentu.

```
mf=["m","f"]; yn=["y","n"]; ev=strtokens("g vg m b vb");
```

Sekarang kita membaca tabel dengan terjemahan ini.

Argumen `tok2`, `tok4` dll. adalah terjemahan dari kolom tabel. Argumen ini tidak ada dalam daftar parameter `readtable()`, jadi Anda perlu menyediakannya dengan ":".

```
MT,hd=readtable("table.dat",tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
```

```
load over statistics;
```

```
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

Titik "." mewakili nilai-nilai, yang tidak tersedia.

Jika kita tidak ingin menentukan token yang akan diterjemahkan terlebih dahulu, 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);
```

`ctok=[2,4,5,7]`: Ini adalah untuk menentukan kolom yang akan diambil yaitu kolom ke-2, ke-4, ke-5, dan ke-7.

```
tok
```

```
m n f y g vg
```

Tabel berisi entri dari file dengan token yang diterjemahkan ke dalam angka.

String khusus NA = "." diartikan sebagai "Tidak Tersedia", dan mendapatkan NAN (bukan angka) di tabel. Terjemahan ini dapat diubah dengan parameter NA, dan NAval.

```
MT[1]
```

```
[1, 1, 30, 2, NAN, 1.8, 2]
```

Berikut isi tabel dengan nomor yang belum 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 memasukkan keluaran `readtable()` ke dalam 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 tabel daftar.

```
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 tabel, melewati baris apa pun dengan nilai `NAN` (". " dalam file), dan indeks kolom, yang berisi nilai-nilai ini.

```
c,i=tablecol(MT,[5,6]);
```

Kita bisa menggunakan ini untuk mengekstrak kolom dari tabel untuk tabel baru.

```
j=[1,5,6]; writetable(MT[i,j],lab=hd[j],ctok=[2],tok=tok)
```

```

Person Evaluation Tip 2 g 1.8 3 g 1.8 6 g 2.8 7 vg 2.8 9 vg 1.8 11 vg 1.8 12
g 1.8 13 vg 1.8 14 g 1.8 15 g 0.8 16 g 2.8 20 vg 1.8 21 m 2.8 22 vg 1.8 23 g 2.8
24 m 1.8

```

Tentu saja, kita perlu mengekstrak tabel itu sendiri dari daftar Tabel dalam kasus ini.

```
MT=Table[1];
```

Tentu saja, kita juga dapat menggunakannya untuk menentukan nilai rata-rata suatu kolom atau nilai statistik lainnya.

```
mean(tablecol(MT,6))
```

```
2.175
```

Fungsi `getstatistics()` mengembalikan elemen dalam vektor, dan jumlahnya. Kami menerapkannya pada nilai "m" dan "f" di kolom kedua tabel kami.

```
xu,count=getstatistics(tablecol(MT,2)); xu, count,
```

```
[1, 3] [12, 13]
```

Kita bisa mencetak hasilnya di tabel baru.

```
writetable(count',labr=tok[xu])
```

```
m 12 f 13
```

Fungsi `selecttable()` mengembalikan tabel baru dengan nilai dalam satu kolom yang dipilih dari vektor indeks. Pertama kita mencari indeks dari dua nilai kita di tabel token.

```
v:=indexof(tok,["g","vg"])
[5, 6]
```

Sekarang kita dapat memilih baris tabel, yang memiliki salah satu nilai v pada baris ke-5.

```
MT1:=MT[selectrows(MT,5,v)]; i:=sortedrows(MT1,5);
```

Sekarang kita dapat mencetak tabel, dengan nilai yang diekstraksi 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 selanjutnya, kami ingin menghubungkan dua kolom tabel. Jadi kita ekstrak kolom 2 dan 4 dan urutkan tabelnya.

```
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 y m y m y m y m y f n f y f y f y f y
f y f y f y f y f y f y f y
```

Dengan `getstatistics()`, kita juga bisa menghubungkan jumlah 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
```

Sebuah tabel dapat ditulis ke file.

```
filename="test.dat"; ... writetable(count,labr=tok[xu1],labc=tok[xu2],file=filename);
```

Kemudian kita bisa 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 filenya.

```
fileremove(filename);
```

Distribusi

Dengan `plot2d`, ada metode yang sangat mudah untuk memplot sebaran data eksperimen.

```
p=normal(1,1000);
plot2d(p,distribution=20,style="
/");
```

```
plot2d("qnormal(x,0,1)",add=1):
```

```
!images/STATISTIKALasigi
```

`p=normal(1,1000)`; digunakan untuk menciptakan 1000 sampel acak yang terdistribusi normal dengan mean (rata-rata) 1 dan standar deviasi 1000.

```
plot2d("qnormal(x,0,1)",add=1);
```

digunakan untuk menambahkan plot dari distribusi normal standar (dengan mean 0 dan standar deviasi 1) ke grafik yang sama. Fungsi `qnormal(x,0,1)` mengacu pada distribusi kumulatif dari variabel acak normal standar. `add=1`

menunjukkan bahwa grafik ini harus ditambahkan ke grafik yang sudah ada, bukan dibuat baru.

Perlu diperhatikan perbedaan antara bar plot (sampel) dan kurva normal (distribusi sebenarnya). Masukkan kembali ketiga perintah untuk melihat hasil pengambilan sampel lainnya.

Berikut adalah perbandingan 10 simulasi dari 1000 nilai terdistribusi normal menggunakan apa yang disebut plot kotak. Plot ini menunjukkan median, kuartil 25, outlier.

```
p=normal(10,1000); boxplot(p):
```

```
!images/STATISTIKA_Lasigi
```

Untuk menghasilkan bilangan bulat acak, Euler memiliki `inrandom`. Mari kita simulasi lemparan dadu dan plot distribusinya.

Kita menggunakan fungsi `getmultiplicities(v,x)`, yang menghitung seberapa sering elemen `v` muncul di `x`. Kemudian kita plot hasilnya menggunakan `kolomplot()`.

```
k=inrandom(1,6000,6); ... columnsplot(getmultiplicities(1:6,k)); ...  
ygrid(1000,color=red):
```

```
!images/STATISTIKA_Lasigi
```

Meskipun `inrandom(n,m,k)` mengembalikan bilangan bulat yang terdistribusi secara seragam dari 1 hingga `k`, distribusi bilangan bulat lainnya dapat digunakan dengan `randpint()`.

Dalam contoh berikut, probabilitas untuk 1,2,3 masing-masing adalah 0,4,0.1,0.5.

```
randpint(1,1000,[0.4,0.1,0.5]); getmultiplicities(1:3,  
[378, 102, 520])
```

Euler dapat menghasilkan nilai acak dari lebih banyak distribusi. Lihat referensinya.

Misalnya, kita mencoba distribusi eksponensial. Variabel acak kontinu `X` dikatakan berdistribusi eksponensial, jika PDF-nya diberikan oleh

with parameter

```
plot2d(randexponential(1,1000,2), distribution):
```

```
!images/STATISTIKA_Lasigi
```

Parameter pertama (1) adalah `lambda`, yang merupakan parameter distribusi eksponensial.

Parameter kedua (1000) menunjukkan jumlah angka acak yang dihasilkan.

Parameter ketiga (2) bisa menunjukkan dimensi atau bentuk output.

Untuk banyak distribusi, Euler dapat menghitung fungsi distribusi dan inversnya.

```
plot2d("normaldis",-4,4):
```

```
!images/STATISTIKA_Lasigi
```

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):
```

```
!images/STATISTIKA_Lasigi
```

Peluang berada di kawasan hijau adalah sebagai berikut.

```
normaldis(5,1,1.5)-normaldis(2,1,1.5)
```

```
0.248662156979
```

Ini dapat dihitung secara numerik dengan integral berikut.

```
gauss("qnormal(x,1,1.5)",2,5)
0.248662156979
```

Mari kita bandingkan distribusi binomial dengan distribusi normal yang mean dan deviasinya sama. Fungsi `invbindis()` menyelesaikan interpolasi linier antara nilai integer.

```
invbindis(0.95,1000,0.5), invnormaldis(0.95,500,0.5sqrt(1000))
525.516721219 526.007419394
```

Fungsi `qdis()` adalah kepadatan distribusi chi-kuadrat. Seperti biasa, Euler memetakan vektor ke fungsi ini. Dengan demikian kita mendapatkan plot semua distribusi chi-kuadrat dengan derajat 5 sampai 30 dengan mudah dengan cara berikut.

```
plot2d("qchidis(x,(5:5:50)')",0,50):
![images/STATISTIKA_Lasigi
```

Euler memiliki fungsi akurat untuk mengevaluasi distribusi. Mari kita periksa `chidis()` dengan integral.

```
Penamaannya mencoba untuk konsisten. Misalnya.,
* distribusi chi-kuadratnya adalah chidis(),
* fungsi kebalikannya adalah invchidis(),
* kepadatannya adalah qchidis().
```

Pelengkap distribusi (ekor atas) adalah `chicdis()`.

```
chidis(1.5,2), integrate("qchidis(x,2)",0,1.5)
0.527633447259 0.527633447259
```

Distribusi Diskrit

Distribusi diskret adalah jenis distribusi probabilitas yang digunakan untuk variabel acak diskret, yaitu variabel yang hanya dapat memiliki nilai tertentu, biasanya dalam bentuk bilangan bulat.

Untuk menentukan distribusi diskrit Anda sendiri, Anda dapat menggunakan metode berikut.

Pertama kita atur 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]
```

Perintah ini menggunakan operator `—` dan `+` untuk membuat nilai dalam variabel `wd`.

`1:6` Ini menghasilkan vektor `[1, 2, 3, 4, 5, 6]`.

`(1:6) + [-0.01, 0.01, 0, 0, 0, 0]`: Operasi ini menambahkan kedua vektor elemen per elemen.

Hasilnya:

```
[1-0.01,2+0.01,3,4,5,6]=[0.99,2.01,3,4,5,6]/6
```

Membagi setiap elemen hasil penjumlahan tadi dengan 6.

Hasilnya:

Artinya dengan probabilitas `wd[i+1]-wd[i]` kita menghasilkan nilai acak `i`.

Ini hampir merupakan distribusi yang seragam. Mari kita tentukan generator nomor acak untuk ini. Fungsi `find(v,x)` mencari nilai `x` pada vektor `v`. Fungsi ini juga berfungsi untuk vektor `x`.

```
function wrongdice (n,m) := find(wd,random(n,m))
```

Kesalahannya sangat halus sehingga kita hanya melihatnya dengan banyak iterasi.

Fungsi `wrongdice` mengembalikan sebuah matriks berukuran  $n \times m$ , di mana setiap elemen dari matriks ini adalah indeks posisi dari elemen `wd` yang paling sesuai (atau mendekati) nilai acak dari `random(n, m)`.

```
columnsplot(getmultiplicities(1:6,wrongdice(1,1000000))):
!images/STATISTIKALasigi
```

Hasil `columnsplot` akan menunjukkan frekuensi relatif dari setiap angka (1 hingga 6), yang memungkinkan Anda untuk melihat apakah distribusi itu merata atau tidak.

Berikut adalah fungsi sederhana untuk memeriksa keseragaman distribusi nilai  $1 \dots K$  dalam  $v$ . Kita menerima hasilnya, jika untuk semua frekuensi

Metode tersebut merupakan metode statistik untuk menguji keseragaman distribusi. Distribusi dianggap seragam jika frekuensi setiap nilai dalam  $v$  mendekati frekuensi ideal  $1/K$ , dengan deviasi yang tidak melebihi batas toleransi.

```
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 ;/prej. Memang fungsinya menolak distribusi seragam.
checkrandom(wrongdice(1,1000000))
```

0

Dan ia menerima generator acak bawaan.

Manual:

- Asumsi dadu, maka peluang setiap sisi =  $1/6$

Dalam 1 juta lemparan maka

\* Frekuensi setiap sisi `fr`. Proporsi tiap sisi = `fr/n` \* Misalkan frekuensi munculnya angka adalah \* [160000, 170000, 180000, 150000, 170000, 170000]

\* Maka proporsi setiap angka: \* [160000, 170000, 180000, 150000, 170000, 170000]  $\frac{1}{1000000} \times [0.16, 0.17, 0.18, 0.15, 0.17, 0.17]$

\* Deviasi maksimum  $\text{fn}/n - 1/K * 1$   $K=\frac{1}{6}=0.1667 * ([0.16, 0.17, 0.18, 0.15, 0.17, 0.17] - 0.1667) * \max(-0.0067, 0.0033, 0.0133, -0.0133, 0.0133, 0.0133)$

\* Bandingkan dengan batas toleransi. \* Batas=  $\text{delta} \frac{1}{\sqrt{n}} = \frac{1}{\sqrt{1000000}} = \frac{1}{1000} = 0.001 * 0.0133 \text{gt}; 0.001$

Hasil 0 di sini mengindikasikan bahwa fungsi `checkrandom` telah menentukan bahwa distribusi tidak seragam.

```
checkrandom(intrandom(1,1000000,6))
```

1

`checkrandom` mengembalikan 1 atau true yang berarti bahwa distribusi dari 1 juta bilangan acak rentang 1 sampai 6 dianggap cukup seragam dalam batas toleransi yang ditetapkan.

Kita dapat menghitung distribusi binomial. Pertama ada `binomialsun()`, yang mengembalikan probabilitas  $i$  atau kurang hit dari  $n$  percobaan.

Misal kita akan menghitung probabilitas dari distribusi binomial di mana terdapat 1000 percobaan (misalnya, 1000 kali pelemparan koin), dengan probabilitas sukses pada setiap percobaan sebesar 0.4, dan kita ingin mengetahui probabilitas mendapatkan tepat 410 sukses.

Secara matematis, ini dihitung dengan rumus:

```

bindis(410,1000,0.4)
0.751401349654
bindis(4,10,0.6)
0.1662386176
Manual:
Secara matematis, ini dihitung dengan rumus:
* Untuk k = 0 * P(X=0)= 100 · (0.6)0 · (0.4)10 ≈ 0.00010
* Untuk k = 1 * P(X = 1)= 101 · (0.6)1 · (0.4)9 ≈ 0.00157
* Untuk k = 2 * P(X = 2)= 102 · (0.6)2 · (0.4)8 ≈ 0.01061
* Untuk k = 3 * P(X = 3)= 103 · (0.6)3 · (0.4)7 ≈ 0.04246
* Untuk k = 4 * P(X = 4)= 104 · (0.6)4 · (0.4)6 ≈ 0.11147 * Maka, *P(X ≤
4) = P(X = 0) + P(X = 1) + P(X = 2) + P(X = 3) + P(X = 4) * P(X ≤ 4) =
0.00010 + 0.00157 + 0.01061 + 0.04246 + 0.11147 * P(X ≤ 4) ≈ 0.1662
Fungsi Beta terbalik digunakan untuk menghitung interval kepercayaan Clopper-
Pearson untuk parameter p. Tingkat defaultnya adalah alfa.
Arti dari interval ini adalah jika p berada di luar interval, hasil pengamatan
410 dalam 1000 jarang terjadi.
clopperpearson(410,1000)
[0.37932, 0.441212]
Perintah berikut adalah cara langsung untuk mendapatkan hasil di atas.
Namun untuk n yang besar, penjumlahan langsungnya tidak akurat dan lambat.
p=0.4; i=0:410; n=1000; sum(bin(n,i)pi(1 - p)(n - i))
0.751401349655
Omong-omong, invbinsum() menghitung kebalikan dari binomialsum().
invbindis(0.75,1000,0.4)
409.932733047
Di Bridge, kami mengasumsikan 5 kartu beredar (dari 52) 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).
2hypergeomsum(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
Merencanakan Data/ Plot Data
Untuk memetakan data, kami mencoba hasil pemilu Jerman sejak tahun
1990, diukur dalam jumlah 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 beberapa bagian, kami menggunakan rangkaian nama.
P:=[ "CDU/CSU", "SPD", "FDP", "Gr", "Li"];
Mari kita cetak persentasenya dengan baik.

```



Pertama kita mengekstrak kolom yang diperlukan. Kolom 3 sampai 7 adalah kursi masing-masing partai, dan kolom 2 adalah jumlah kursi seluruhnya. Kolom 1 adalah tahun pemilihan.

```
BT:=BW[,3:7]; BT:=BT/sum(BT); YT:=BW[,1]';
```

Kemudian statistiknya kita cetak dalam bentuk tabel. Kami menggunakan nama sebagai header kolom, dan tahun sebagai header untuk baris. Lebar default untuk kolom adalah `wc=10`, tetapi kami lebih memilih keluaran yang lebih padat. Kolom akan diperluas untuk label kolom, jika perlu.

```
writetable(BT100,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 menjumlahkan persentase dua partai besar yang menunjukkan bahwa partai-partai kecil berhasil 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. Kami menggunakannya untuk menampilkan garis dan titik secara bersamaan. Alternatifnya adalah memanggil `plot2d` dua kali dengan `gt;add`.

```
statplot(YT,BT1,"b"):
```

```
!images/STATISTIKALasigi
```

Tentukan beberapa warna untuk setiap pesta.

```
CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.8,0,0)];
```

Sekarang kita bisa memplot hasil pemilu 2009 dan perubahannya menjadi satu plot dengan menggunakan gambar. Kita dapat menambahkan vektor kolom ke setiap plot.

```
figure(2,1); ... figure(1); columnsplot(BW[6,3:7],P,color=CP); ... figure(2); columnsplot(BW[6,3:7]-BW[5,3:7],P,color=CP); ... figure(0):
```

```
!images/STATISTIKALasigi
```

Plot data menggabungkan deretan data statistik dalam satu plot.

```
J:=BW[,1]'; DP:=BW[,3:7]'; ... dataplot(YT,BT',color=CP); ... label-box(P,color=CP,styles="[]", points,w=0.2,x=0.3,y=0.4):
```

```
!images/STATISTIKALasigi
```

Plot kolom 3D memperlihatkan baris data statistik dalam bentuk kolom.

Kami memberikan label untuk baris dan kolom. sudut adalah sudut pandang.

```
columnsplot3d(BT,scols=P,srows=YT, ... angle=30°,ccols=CP):
```

```
!images/STATISTIKALasigi
```

Representasi lainnya adalah plot mosaik. Perhatikan bahwa kolom plot mewakili kolom matriks di sini. Karena panjang label CDU/CSU, kami mengambal jendela yang lebih kecil dari biasanya.

```
shrinkwindow( smaller); ... mosaicplot(BT',srows=YT,scols=P,color=CP,style="");
... shrinkwindow():
```

```
!images/STATISTIKALasigi
```

Kita juga bisa membuat diagram lingkaran. Karena hitam dan kuning membentuk koalisi, kami menyusun ulang elemen-elemennya.

```
i=[1,3,5,4,2]; piechart(BW[6,3:7][i],color=CP[i],lab=P[i]):
```

```

!images/STATISTIKA_Lasigi
Ini adalah jenis plot lainnya.
starplot(normal(1,10)+4,lab=1:10, rays):
!images/STATISTIKA_Lasigi
Beberapa plot di plot2d bagus untuk statika. Berikut adalah plot impuls
dari data acak, terdistribusi secara seragam di [0,1].
plot2d(makeimpulse(1:10,random(1,10)), bar):
!images/STATISTIKA_Lasigi
Namun untuk data yang terdistribusi secara eksponensial, kita mungkin
memerlukan plot logaritmik.
logimpulseplot(1:10,-log(random(1,10))10):
!images/STATISTIKA_Lasigi
Fungsi Columnplot() lebih mudah digunakan, karena hanya memerlukan vek-
tor nilai. Selain itu, ia dapat mengatur labelnya ke apa pun yang kita inginkan,
kami telah mendemonstrasikannya di tutorial ini.
Ini adalah aplikasi lain, di mana kita menghitung karakter dalam sebuah
kalimat dan membuat 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; ...
columnplot(x,lab=cw,width=0.05):
!images/STATISTIKA_Lasigi
Dimungkinkan juga untuk mengatur sumbu secara manual.
n=10; p=0.4; i=0:n; x=bin(n,i)p^i(1-p)^(n-i); ... columnplot(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(["Binomialdistribution", "with p =
0.4"]):
!images/STATISTIKA_Lasigi
Berikut ini cara memplot frekuensi bilangan dalam suatu vektor.
Kami membuat vektor bilangan acak bilangan bulat 1 hingga 6.
v:=inrandom(1,10,10)
[8, 5, 8, 8, 6, 8, 8, 3, 5, 5]
Kemudian ekstrak nomor unik di v.
vu:=unique(v)
[3, 5, 6, 8]
Dan plot frekuensi dalam plot kolom.
columnplot(getmultiplicities(vu,v),lab=vu,style="/"):
!images/STATISTIKA_Lasigi
Kami ingin mendemonstrasikan fungsi distribusi nilai empiris.
x=normal(1,20);
Fungsi empdist(x,vs) memerlukan array nilai yang diurutkan. Jadi kita
harus mengurutkan x sebelum kita dapat menggunakannya.
xs=sort(x);
Kemudian kita plot distribusi empiris dan beberapa batang kepadatan ke
dalam satu plot. Alih-alih plot batang untuk distribusi kali ini kami menggu-
nakan plot gigi gergaji.

```

```
figure(2,1); ... figure(1); plot2d("empdist",-4,4;xs); ... figure(2); plot2d(histo(x,v=-
4:0.2:4,i;bar)); ... figure(0):
```

```
!images/STATISTIKALasigi
```

Plot sebar mudah dilakukan di Euler dengan plot titik biasa. Grafik berikut menunjukkan bahwa X dan X+Y jelas berkorelasi positif.

```
x=normal(1,100); plot2d(x,x+rotright(x), points,style="..");
```

```
!images/STATISTIKALasigi
```

Seringkali kita ingin membandingkan dua sampel dengan distribusi yang berbeda. Hal ini dapat dilakukan dengan plot kuantil-kuantil.

Untuk pengujiannya, 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):
```

```
!images/STATISTIKALasigi
```

Plot tersebut dengan jelas menunjukkan bahwa nilai terdistribusi normal cenderung lebih kecil di ujung ekstrim.

Jika kita mempunyai dua distribusi yang ukurannya berbeda, kita dapat memperluas distribusi yang lebih kecil atau mengecilkan distribusi yang lebih besar. Fungsi berikut ini baik untuk keduanya. Dibutuhkan 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):
```

```
!images/STATISTIKALasigi
```

Regresi dan Korelasi

Regresi linier dapat dilakukan dengan fungsi polyfit() atau berbagai fungsi fit.

Sebagai permulaan kita menemukan garis regresi untuk data univariat dengan polyfit(x,y,1).

```
x=1:10; y=[2,3,1,5,6,3,7,8,9,8]; writetable(x'—y',labc=["x","y"])
```

```
x y 1 2 2 3 3 1 4 5 5 6 6 3 7 7 8 8 9 9 10 8
```

Kami ingin membandingkan kecocokan yang tidak berbobot dan berbobot. Pertama koefisien kecocokan linier.

```
p=polyfit(x,y,1)
[0.733333, 0.812121]
```

Regresi linear dapat ditulis dalam bentuk:

dengan

Kita hitung:

Maka:

Jadi, b, m = 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 memasukkan semuanya ke dalam satu plot untuk 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,sty
"); ... plot2d("evalpoly(x,pw)",5,10, add,color=red,style="-"); ... figure(2);
plot2d(w,1,10, filled,style="/",fillcolor=red,xl=w); ... figure(0):
!images/STATISTIKALasigi
```

Contoh lain kita membaca survei siswa, usia mereka, usia orang tua mereka dan jumlah saudara kandung dari sebuah file.

Tabel ini berisi "m" dan "f" di kolom kedua. Kami menggunakan variabel tok2 untuk mengatur terjemahan yang tepat alih-alih membiarkan readtable() mengumpulkan terjemahannya.

```
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 bergantung satu sama lain? Kesan pertama muncul dari plot sebar berpasangan.

```
scatterplots(tablecol(MS,3:5),hd[3:5]):
!images/STATISTIKALasigi
Jelas terlihat 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  $s=17+0,74t$ , dengan  $t$  adalah umur ibu dan  $s$  adalah umur ayah. Perbedaan usia mungkin sedikit bergantung pada usia, tapi tidak terlalu banyak.

Sebaliknya, kami mencurigai fungsi seperti  $s=a+t$ . Maka  $a$  adalah mean dari  $s-t$ . Ini adalah perbedaan usia rata-rata antara ayah dan ibu.

```
da:=mean(cs[2]-cs[1])
3.65
Mari kita plot ini menjadi satu plot sebar.
plot2d(cs[1],cs[2], points); ... plot2d("evalpoly(x,ps)",color=red,style=".", add);
... plot2d("x+da",color=blue, add):
!images/STATISTIKALasigi
```

Berikut adalah plot kotak dari dua zaman tersebut. Ini hanya menunjukkan, bahwa usianya berbeda-beda.

```
boxplot(cs,["mothers","fathers"]):
!images/STATISTIKALasigi
Menariknya, perbedaan median tidak sebesar perbedaan mean.
median(cs[2])-median(cs[1])
1.5
```

Koefisien korelasi menunjukkan korelasi positif.

```
Koefisien korelasi menunjukkan korelasi positif.correl(cs[1],cs[2])
```

Variable Koefisien not found! Error in: Koefisien korelasi menunjukkan korelasi positif.correl(cs[1],c ...

Korelasi pangkat merupakan ukuran keteraturan yang sama pada kedua vektor. Hal ini juga cukup positif.

```
rankcorrel(cs[1],cs[2])
```

```
0.758925292358
```

Membuat Fungsi baru

Tentu saja, bahasa EMT dapat digunakan untuk memprogram fungsi-fungsi baru. Misalnya, kita mendefinisikan fungsi skewness.

$$sk(x) = \sqrt{n} \sum_i (x_i - m)^3 \left( \sum_i (x_i - m)^2 \right)^{3/2}$$

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 <
```

/pre > *SepertiyangAndalihat,kitadapatdenganmudahmenggunakanbahasamatriksuntukmendapatkanimpl*

```
  data=normal(20); skew(normal(10))
```

```
0.0671423500544
```

Berikut adalah fungsi lainnya, yang disebut koefisien skewness Pearson.

```
function skew1 (x) := 3(mean(x)-median(x))/dev(x)
```

```
skew1(data)
```

```
0.622743747628
```

Simulasi Monte Carlo

Kita simulasikan variabel acak berdistribusi normal 1000-5 sebanyak sejuta kali. Untuk ini, kita gunakan fungsi normal(m,n), yang menghasilkan matriks nilai berdistribusi 0-1, atau normal(n) yang secara default bernilai m=1.

```
n=1000000; x=normal(n)
```

```
[-0.661822, 1.14053, -0.0725198, -0.653134, 0.338853, 0.158539, -1.33566, -
1.21098, 0.252185, 0.750436, -0.335005, 0.79995, 0.237049, 0.10056, -0.909771, -
0.896389, 0.807295, 0.420089, -2.60022, -0.334883, 0.563393, -1.51976, -0.618486,
-0.757591, -1.5655, -1.14804, -1.62101, -0.767098, 1.41355, 0.28399, 1.31699,
0.325665, 0.123923, -0.301708, 0.224152, 1.61033, -0.928959, 0.112323, 2.06755,
-0.415012, 1.227, -0.392912, -0.82682, -1.37726, 1.09679, -2.83431, 0.310441,
1.333, -0.774371, 1.51449, 0.0260218, 0.244161, -1.25954, 0.833535, -1.05672, -
3.99147, -1.70256, 0.620494, -0.98936, -0.868208, 0.344441, 0.926404, 0.00735312,
-1.07317, 0.26722, 0.469612, -1.44652, -0.0850004, -1.22639, -0.120585, -0.498201,
0.793201, -2.71956, -0.832553, -1.50601, -0.473103, -0.857245, -0.397193, -0.368307,
-1.08578, -1.60682, -0.525828, 0.00660302, -2.44655, 0.761081, -0.0405162, -
0.440874, -0.733893, 0.275494, 1.14203, 0.0210417, -0.418046, -0.375807, -1.4251,
-0.859738, 0.949762, -0.774583, 0.726568, 1.42364, -0.614302, 0.47965, -0.44265,
0.128351, -0.696414, 0.497947, 0.407174, -1.3959, -0.0367871, 0.6379, -1.03524, -
0.797854, 0.490069, 0.521037, 0.526266, 0.129707, 0.0266106, -0.734832, -0.616116,
-0.675762, -1.81443, ... ]
```

```
n=1000000; x=normal(n)5+1000
```

```
[1001.73, 1005.57, 995.544, 1004.48, 996.802, 1004.15, 1005.39, 997.653, 1002.83,
1005.25, 998.235, 994.648, 998.584, 1000.88, 998.811, 994.607, 995.101, 995.2,
1002.16, 1009.75, 1000.77, 999.994, 1011.51, 1006.36, 990.598, 1007.2, 1000.18,
```

999.05, 1002.22, 1000.08, 993.234, 1000.81, 1004.78, 1002.42, 993.949, 1002.01, 995.443, 994.59, 1008.39, 1000.62, 995.42, 1000.23, 998.282, 998.84, 992.97, 1000.81, 993.34, 1003.17, 1001.89, 999.305, 1000.63, 992.78, 989.389, 998.162, 1001.79, 1008.18, 1003.34, 1001.35, 1001.08, 998.659, 998.586, 1000.31, 994.955, 994.737, 988.751, 991.572, 995.534, 999.851, 1004.17, 997.659, 993.513, 996.48, 999.944, 996.108, 998.717, 1002.66, 1001.39, 1003.17, 992.563, 998.983, 994.664, 997.407, 1001.87, 999.54, 993.469, 999.279, 1000.47, 996.123, 1000.8, 1003.22, 993.011, 998.241, 1000.07, 1001.6, 1002.9, 1004.22, 1002.22, 1005.54, 999.447, 1002.94, 1003.54, 999.995, 996.819, 1001.89, 1003.54, 1005.62, 1002.17, 998.897, 1005.83, 1006.59, 1010.61, 996.573, 999.918, 1001.98, 1002.8, 1003.21, 996.516, 1008.32, 997.694, 998.671, 995.241, 1007.62, 989.539, 1008.12, 1006.46, 1002.18, 1001.9, 1004.22, 1001.93, 1002.28, 994.052, 992.231, 999.155, 997.39, 996.867, 998.962, 1002.89, 1005.8, 1003.44, 1006.92, ... ]

terdapat juga fungsi `randnormal(n,m,mean,dev)`, yang dapat kita gunakan. Fungsi ini mematuhi skema penamaan "rand..." untuk generator acak.

```
n=1000000; x=randnormal(1,n,1000,5)
[1003.03, 1006.05, 999.733, 999.09, 1005.97, 1003.15, 998.16, 989.205, 1002.45,
996.641, 1008.69, 1006.67, 992.93, 1003.96, 1002.56, 997.29, 995.598, 995.616,
1000.14, 1002.68, 984.552, 1004.04, 1008.83, 994.196, 991.508, 1003.67, 1000.44,
1001.38, 996.971, 1001.32, 998.596, 995.479, 1002.12, 990.516, 997.716, 1008.15,
1000.37, 1001.07, 996.014, 1001.07, 1003.76, 996.41, 992.867, 998.441, 993.741,
997.669, 1007.58, 1005.19, 997.211, 1000.75, 994.023, 994.398, 998.096, 994.683,
994.84, 1001.44, 1003.19, 987.72, 1003.07, 1008.01, 1009.52, 1007.39, 997.166,
1007.7, 987.807, 1002.73, 1003.18, 1002.86, 1002.75, 1001.91, 995.783, 998.36,
1002.57, 994.433, 1006.56, 1002.42, 997.431, 1005.47, 992.773, 1007.92, 1007.36,
999.482, 1003.88, 998.68, 1012.77, 1008.83, 1002.04, 1003.13, 1009.78, 997.305,
1003.33, 1001.45, 1004.12, 999.9, 1002.25, 996.054, 994.309, 1006.29, 999.89,
1003.39, 995.42, 990.872, 1000.26, 994.472, 997.125, 998.633, 994.745, 1001.02,
996.873, 1001.29, 999.597, 999.643, 995.944, 995.697, 996.605, 1012.43, 1005.57,
1005.33, 1005.1, 1009.44, 999.924, 1008.11, 999.458, 1000.14, 993.022, 1004.28,
1004.84, 995.023, 998.484, 996.675, 1002.69, 1003.4, 993.025, 998.71, 1000.25,
997.502, 995.476, 997.965, 1005.47, 1003.73, ... ]
```

10 nilai pertama x adalah

```
x[1:10]
[1003.03, 1006.05, 999.733, 999.09, 1005.97, 1003.15, 998.16, 989.205, 1002.45,
996.641]
```

Distribusi dapat kita plot dengan flag `gt;distribution` dari `plot2d`.

```
plot2d(x, distribution); ... plot2d("qnormal(x,1000,5)",color=red,thickness=2, add):
![images/STATISTIKA_Lasigi
```

kita juga dapat mengatur jumlah interval untuk distribusi menjadi 100. Kemudian kita akan melihat seberapa dekat kecocokan distribusi yang diamati dan distribusi yang sebenarnya. Bagaimanapun, kita telah menghasilkan satu juta kejadian.

```
plot2d(x,distribution=100); ... plot2d("qnormal(x,1000,5)",color=red,thickness=2, add):
![images/STATISTIKA_Lasigi
```

kita dapat menghitung nilai rata-rata simulasi dan deviasinya harus sangat dekat dengan nilai yang diharapkan.

```
mean(x), dev(x)
999.996930441 4.9987603217
rumus nilai rata rata
```

$$mean = \frac{\sum x_i}{n}$$

```
xm=sum(x)/n
```

```
999.996930441
```

Rumus simpangan percobaannya (deviasi)

```
sqrt(sum((x-xm)^2/(n-1)))
```

```
4.9987603217
```

Perhatikan bahwa x-xm adalah vektor nilai yang dikoreksi, di mana xm dikurangi dari semua elemen vektor x.

Berikut adalah 10 nilai pertama x-xm.

```
short (x-xm)[1:10]
```

```
[3.0373, 6.054, -0.26415, -0.90714, 5.9728, 3.1492, -1.837, -10.792, 2.4518, -3.3556]
```

Dengan menggunakan bahasa matriks, kita dapat dengan mudah menjawab pertanyaan lainnya. Misalnya, kita ingin menghitung proporsi x yang melebihi 1015.

Ekspresi `xgt;=1015` menghasilkan vektor 1 dan 0. Menjumlahkan vektor ini menghasilkan jumlah kali `x[i]gt;=1015` terjadi.

```
sum(x =1015)/n
```

```
0.001339
```

Probabilitas yang diharapkan dari hal ini dapat dihitung dengan fungsi `normaldis(x)`. sehingga,

dimana X terdistribusi secara normal m-s.

```
1-normaldis(1015,1000,5)
```

```
0.00134989803163
```

cara kerja `gt;distribution` flag dari `plot2d` adalah menggunakan fungsi `histo(x)`, yang menghasilkan histogram frekuensi nilai dalam x. Fungsi ini mengembalikan batas interval dan jumlah dalam interval ini. Kami menormalkan jumlah untuk mendapatkan frekuensi.

```
t,s=histo(x,40); plot2d(t,s/n, bar):
```

```
!images/STATISTIKALasigi
```

Fungsi `histo()` juga dapat menghitung frekuensi dalam interval yang diberikan.

```
t,s=histo(x,v=[950,980,990,1010,1020,1050]); t, s,
```

```
[950, 980, 990, 1010, 1020, 1050] [25, 22796, 954608, 22544, 27]
```

hasil tersebut merupakan semua nilai acak yang berada antara 950 dan 1050.

menghitung total jumlah nilai dalam s, yang sama dengan total jumlah elemen dalam x

```
sum(s)
```

```
1000000
```

kita akan mensimulasikan 1000 kali lemparan 3 dadu, dan menanyakan pembagian jumlahnya.

```
ds=sum(intrandom(1000,3,6)); fs=getmultiplicities(3:18,ds)
[5, 16, 18, 47, 79, 95, 135, 127, 113, 120, 97, 69, 43, 21, 12, 3]
kita akan plot hasil tersebut
columnspot(fs,lab=3:18):
!images/STATISTIKALasigi
```

kita akan menggunakan rekursi tingkat lanjut.

Fungsi berikut menghitung banyaknya cara bilangan k dapat direpresentasikan sebagai jumlah dari n bilangan dalam rentang 1 sampai m.

```
function map countways (k; n, m) ...
if n==1 then return k; k=1 k=m else sum=0; loop 1 to m; sum=sum+countways(k-
,n-1,m); end; return sum; end; endfunction ;/pre; Berikut hasil pelemparan
dadu sebanyak lima kali.
```

```
countways(5:25,5,5)
[1, 5, 15, 35, 70, 121, 185, 255, 320, 365, 381, 365, 320, 255, 185, 121, 70, 35,
15, 5, 1]
```

```
cw=countways(3:18,3,6)
[1, 3, 6, 10, 15, 21, 25, 27, 27, 25, 21, 15, 10, 6, 3, 1]
Kita akan menambahkan nilai yang diharapkan ke plot.
plot2d(cw/631000, add); plot2d(cw/631000, points, add) :
```

```
!images/STATISTIKALasigi
```

Untuk simulasi lain, deviasi nilai rata-rata n 0-1-variabel acak terdistribusi normal adalah  $1/\sqrt{n}$ .

```
longformat; 1/sqrt(10)
0.316227766017
```

Mari kita periksa ini dengan simulasi. Kami menghasilkan 10.000 kali 10 vektor acak.

```
M=normal(10000,10); dev(mean(M)')
0.319361781348
plot2d(mean(M)', distribution):
!images/STATISTIKALasigi
```

Median dari 10 bilangan acak berdistribusi normal 0-1 mempunyai deviasi yang lebih besar.

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 langkah ke-n dari proses ini bersama dengan nilai yang diharapkan berwarna 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):
!images/STATISTIKALasigi
```

Uji Chi-Kuadrat

uji chi-kuadrat adalah alat penting dalam statistik. Di Euler, banyak tes yang diterapkan. Semua pengujian ini mengembalikan kesalahan yang kita terima jika kita menolak hipotesis nol.



Misalnya, kami menguji lemparan dadu untuk distribusi yang seragam. Pada 600 kali lemparan, kami mendapatkan nilai berikut, yang kami masukkan ke dalam uji chi-kuadrat.

```
chitest([90,103,114,101,103,89],dup(100,6)')
0.498830517952
```

Ini adalah nilai p-value dari uji chi-kuadrat

Uji chi-kuadrat juga memiliki mode yang menggunakan simulasi Monte Carlo untuk menguji statistiknya, menggunakan Parameter `gt;p` menafsirkan vektor `y` sebagai vektor probabilitas.

```
chitest([90,103,114,101,103,89],dup(1/6,6)', p, montecarlo)
0.501
```

Ini adalah p-value dari uji chi-kuadrat menggunakan pendekatan Monte Carlo. Dengan simulasi Monte Carlo, kita memperoleh p-value yang mirip dengan uji chi-kuadrat standar (0,4988 di uji pertama)

Selanjutnya kita menghasilkan 1000 lemparan dadu menggunakan generator angka acak, dan melakukan tes yang sama.

```
n=1000; t=random([1,n6]); chitest(count(t6,6),dup(n,6)')
0.555478672553
```

Mari kita uji nilai rata-rata 100 dengan uji-t.

```
s=200+normal([1,100])10; ... ttest(mean(s),dev(s),100,200)
0.121802914522
```

Fungsi `ttest()` memerlukan nilai mean, deviasi, jumlah data, dan nilai mean yang akan diuji.

Sekarang mari kita periksa dua pengukuran untuk mean yang sama. Kami menolak hipotesis bahwa keduanya mempunyai mean yang sama, jika hasilnya `lt;0,05`.

```
tcomparedata(normal(1,10),normal(1,10))
0.435203134071
```

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)
0.00404956559836
```

Menambah nilai 2 ke salah satu distribusi menyebabkan p-value menjadi sangat kecil.

Pada contoh berikutnya, kita membuat 20 lemparan dadu acak sebanyak 100 kali dan menghitung yang ada di dalamnya. Rata-rata harus ada  $20/6=3,3$ .

```
R=random(100,20); R=sum(R6j=1)'; mean(R)
3.46
```

Sekarang kita bandingkan jumlah satuan dengan distribusi binomial. Pertama kita plot distribusinya.

```
plot2d(R,distribution=max(R)+1,even=1,style="
/");
```

```
![images/STATISTIKALasigi
```

kita akan Menghitung frekuensi kemunculan setiap jumlah angka "1" dalam 20 lemparan dadu acak yang telah dilakukan 100 kali

```
t=count(R,21);
```

Kemudian kami menghitung nilai yang diharapkan.  
 $n=0:20$ ;  $b=\text{bin}(20,n)(1/6)^n(5/6)^{(20-n)}100$ ;  
 Kita harus mengumpulkan beberapa angka untuk mendapatkan kategori yang cukup besar.  
 $t1=\text{sum}(t[1:2])—t[3:7]—\text{sum}(t[8:21]); \dots$   $b1=\text{sum}(b[1:2])—b[3:7]—\text{sum}(b[8:21]);$   
 Uji chi-square menolak hipotesis bahwa distribusi kita merupakan distribusi binomial, jika hasilnya  $lt;0,05$ .  
 $\text{chitest}(t1,b1)$   
 0.401747438977  
 Contoh berikut berisi hasil dua kelompok orang (misalnya laki-laki dan perempuan) yang memilih satu dari enam partai.  
 $A=[23,37,43,52,64,74;27,39,41,49,63,76]; \dots$   $\text{writetable}(A,wc=6,labr=["m","f"],labc=1:6)$   
 1 2 3 4 5 6 m 23 37 43 52 64 74 f 27 39 41 49 63 76  
 Kita akan menguji independensi suara dari jenis kelamin.  
 $\text{tabletest}(A)$   
 0.990701632326  
 Berikut ini adalah tabel yang diharapkan, jika kita mengasumsikan frekuensi pemungutan suara yang diamati.  
 $\text{writetable}(\text{expectedtable}(A),wc=6,dc=1,labr=["m","f"],labc=1:6)$   
 1 2 3 4 5 6 m 24.9 37.9 41.9 50.3 63.3 74.7 f 25.1 38.1 42.1 50.7 63.7 75.3  
 Kita dapat menghitung koefisien kontingensi yang dikoreksi. Karena sangat mendekati 0, kami menyimpulkan bahwa pemungutan suara tidak bergantung pada jenis kelamin.  
 $\text{contingency}(A)$   
 0.0427225484717  
 Uji F  
 Selanjutnya kita menggunakan analisis varians (uji F) untuk menguji tiga sampel data yang berdistribusi normal untuk nilai mean yang sama. Metode tersebut disebut ANOVA (analisis varians). Di Euler, fungsi  $\text{varanalysis}()$  digunakan.  
 $x1=[109,111,98,119,91,118,109,99,115,109,94]; \text{mean}(x1),$   
 106.545454545  
 $x2=[120,124,115,139,114,110,113,120,117]; \text{mean}(x2),$   
 119.111111111  
 $x3=[120,112,115,110,105,134,105,130,121,111]; \text{mean}(x3)$   
 116.3  
 $\text{varanalysis}(x1,x2,x3)$   
 0.0138048221371  
 Dengan p-value sebesar 0.0138 (1,38% bahwa ketiga sampel memiliki mean yang sama pada tingkat signifikansi 5% terdapat perbedaan yang signifikan antara mean dari setidaknya satu sampel.  
 Ada juga uji median, yang menolak sampel data dengan distribusi rata-rata yang berbeda, menguji median dari sampel yang disatukan.  
 $a=[56,66,68,49,61,53,45,58,54]$   
 $[56, 66, 68, 49, 61, 53, 45, 58, 54]$   
 $b=[72,81,51,73,69,78,59,67,65,71,68,71]$

```
[72, 81, 51, 73, 69, 78, 59, 67, 65, 71, 68, 71]
```

```
mediantest(a,b)
```

```
0.0241724220052
```

Tes kesetaraan lainnya adalah tes peringkat. Ini jauh lebih tajam daripada tes median.

```
ranktest(a,b)
```

```
0.00199969612469
```

Pada contoh berikut, kedua distribusi mempunyai mean yang sama.

```
ranktest(random(1,100),random(1,50)3-1)
```

```
0.495228684732
```

ini menunjukkan bahwa perbedaan tidak cukup signifikan pada tingkat signifikansi 5 median yang sama tidak dapat ditolak.

Sekarang mari kita coba mensimulasikan dua perlakuan 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];
```

Tes signum memutuskan, apakah a lebih baik dari b.

```
signtest(a,b)
```

```
0.0546875
```

Ini kesalahan yang terlalu besar untuk menolak hipotesis. Kita tidak dapat menolak bahwa a sama baiknya dengan b, Karena  $p > 0.05$ .

Uji Wilcoxon lebih tajam dibandingkan uji ini, namun mengandalkan nilai kuantitatif perbedaannya.

```
wilcoxon(a,b)
```

```
0.0296680599405
```

Mari kita coba dua tes lagi menggunakan rangkaian yang dihasilkan.

```
wilcoxon(normal(1,20),normal(1,20)-1)
```

```
0.0068706451766
```

ini menunjukkan bahwa ada perbedaan signifikan antara kedua sampel pada tingkat signifikansi 5

```
wilcoxon(normal(1,20),normal(1,20))
```

```
0.559353645673
```

hasil ini jauh di atas 0.05, sehingga kita tidak bisa menolak hipotesis bahwa kedua sampel berasal dari distribusi yang sama.

Angka Acak

Berikut ini adalah pengujian pembangkit bilangan acak. Euler menggunakan generator yang sangat bagus, jadi kita tidak perlu mengharapkan adanya masalah.

Pertama kita menghasilkan sepuluh juta angka acak di  $[0,1]$ .

```
n:=10000000; r:=random(1,n);
```

Selanjutnya kita hitung jarak antara dua angka yang kurang dari 0,05.

```
a:=0.05; d:=differences(nonzeros(rja));
```

Terakhir, kami memplot berapa kali, setiap jarak terjadi, dan membandingkannya dengan nilai yang diharapkan.

```
m=getmultiplicities(1:100,d); plot2d(m); ... plot2d("n(1-a)(x-1)a2", color = red, add) :
```

```

!images/STATISTIKA_Lasigi
Hapus datanya.
remvalue n;
Kami ingin menghitung nilai rata-rata dan simpangan baku yang diukur.
M=[1000,1004,998,997,1002,1001,998,1004,998,997]; ... mean(M), dev(M),
999.9 2.72641400622
Kita dapat membuat diagram kotak dan kumis untuk data tersebut. Dalam
kasus kita, tidak ada outlier.
boxplot(M):
!images/STATISTIKA_Lasigi
Kami menghitung probabilitas bahwa suatu nilai lebih besar dari 1005, den-
gan asumsi nilai terukur dan distribusi normal.
Semua fungsi untuk distribusi dalam Euler diakhiri dengan ...dis dan menghi-
tung distribusi probabilitas kumulatif (CPF).
Kami mencetak hasilnya dalam fungsi cetak.
print((1-normaldis(1005,mean(M),dev(M)))100,2,unit="
3.07
Untuk contoh berikutnya, kami mengasumsikan jumlah pria berikut dalam
rentang ukuran tertentu.
r=155.5:4:187.5; v=[22,71,136,169,139,71,32,8];
Berikut adalah plot distribusinya.
plot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style="
/"):
!images/STATISTIKA_Lasigi
Kita dapat memasukkan data mentah tersebut ke dalam tabel.
Tabel adalah metode untuk menyimpan data statistik. Tabel kita harus
berisi tiga kolom: Awal rentang, akhir rentang, jumlah orang dalam rentang.
Tabel dapat dicetak dengan tajuk. Kita menggunakan vektor string untuk
mengatur tajuk.
T:=r[1:8]' — r[2:9]' — v'; writetable(T,labc=["from","to","count"])
from to count 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 memerlukan nilai rata-rata dan statistik ukuran lainnya, kita perlu
menghitung titik tengah rentang. Kita dapat menggunakan dua kolom pertama
tabel kita untuk ini.
(T[,1]+T[,2])/2
157.5 161.5 165.5 169.5 173.5 177.5 181.5 185.5
Namun lebih mudah untuk melipat rentang dengan vektor [1/2,1/2].
l=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(l,v); m, d,
169.901234568 5.98912964449
Mari kita tambahkan distribusi normal nilai-nilai tersebut ke plot.
plot2d("qnormal(x,m,d)sum(v)4", ... xmin=min(r),xmax=max(r),thickness=3,add=1):

```

!images/STATISTIKA<sub>Lasigi</sub>

Pengantar untuk Pengguna Proyek R

Jelasnya, EMT tidak bersaing dengan R sebagai 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.

Notebook ini cocok untuk Anda yang sudah familiar dengan R, namun perlu mengetahui perbedaan sintaksis EMT dan R. Kami mencoba memberikan gambaran umum tentang hal-hal yang sudah jelas dan kurang jelas yang perlu Anda ketahui.

Selain itu, kami mencari cara untuk bertukar data antara kedua sistem.

Note that this is a work in progress.

Sintaks Dasar

Hal pertama yang Anda pelajari di R adalah membuat vektor. Dalam EMT, perbedaan utamanya adalah operator `:` dapat mengambil ukuran langkah. Selain itu, ia mempunyai 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]
```

```
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()` di R. Kita dapat menulis fungsi ini di EMT.

```
function seq(a,b,c) := a:b:c; ... seq(0,-0.1,-1)
```

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

```
function seq(a,b,c) := a:b:c; ... seq(0,-0.5,-5)
```

```
[0, -0.5, -1, -1.5, -2, -2.5, -3, -3.5, -4, -4.5, -5]
```

```
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]
```

Fungsi `rep()` dari R tidak ada di EMT. Untuk masukan vektor dapat dituliskan sebagai berikut.

Perhatikan bahwa `"="` atau `":="` digunakan untuk tugas. Operator `"-gt;"` digunakan untuk satuan dalam EMT.

```
125km - " miles"
```

```
77.6713990297 miles
```

Operator `"lt;-"` untuk penugasan memang bukan ide yang baik untuk R.

tetapi di EMT operator `"lt;-"` itu bukan penugasan melainkan perbandingan

Berikut ini akan membandingkan `a` dan `-4` di EMT.

```
a:=2; a<-4
```

```
0
```

EMT dan R memiliki vektor bertipe boolean. Namun dalam EMT, angka 0 dan 1 digunakan untuk mewakili salah dan benar. Di R, nilai benar dan salah tetap bisa digunakan dalam aritmatika biasa seperti di EMT.

```
x<5,
```

```
[0, 0, 1, 0, 0] [0, 0, 3.1, 0, 0]
```

EMT memunculkan kesalahan atau menghasilkan NAN tergantung pada tanda `"kesalahan"`.

```
errors off; 0/0, isNAN(sqrt(-1)), errors on;
```

```
NAN 1
```

Stringnya sama di R dan EMT. Keduanya berada di lokal saat ini, bukan di Unicode.

Di R ada paket untuk Unicode. Di EMT, string dapat berupa string Unicode. String unicode dapat diterjemahkan ke pengkodean lokal dan sebaliknya. Selain itu, u"..." dapat berisi entitas HTML.

```
u"169; Reneacut; Grothmann"
```

```
© René Grothmann
```

```
karakter khusus (hak cipta © dan karakter aksen é),  
chartoutf([480])
```

Berikut ini mungkin tidak ditampilkan dengan benar pada sistem sebagai A dengan titik dan garis di atasnya. Itu tergantung pada font yang Anda gunakan.

Penggabungan string dilakukan dengan "+" atau "—". Penggabungan ini akan menghasilkan string tunggal, dan angka yang digabungkan akan dikonversi otomatis ke format string. Ini dapat mencakup angka, yang akan dicetak dalam format saat ini.

```
"pi = "+pi
```

```
pi = 3.14159265359
```

```
Pengindeksan
```

Seringkali, ini akan berfungsi seperti di R.

Namun EMT akan menafsirkan indeks negatif dari belakang vektor, sementara R menafsirkan 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
```

```
x, x[1:5], x[-3]
```

```
[10.4, 5.6, 3.1, 6.4, 21.7] [10.4, 5.6, 3.1, 6.4, 21.7] 3.1
```

Untuk meniru perilaku R di EMT, kita dapat menggunakan fungsi drop(x,n)

```
drop(x,2)
```

```
[10.4, 3.1, 6.4, 21.7]
```

Vektor logika tidak diperlakukan berbeda sebagai indeks di EMT, berbeda dengan R. Anda perlu mengekstrak elemen 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]
```

```
Tipe Data
```

EMT memiliki lebih banyak tipe data tetap daripada R. Jelasnya, di R terdapat vektor yang berkembang. Anda dapat mengatur vektor numerik kosong v dan memberikan nilai ke elemen v[17]. Hal ini tidak mungkin dilakukan di EMT.

Berikut ini agak tidak efisien.

```
v=[]; for i=1 to 10000; v=v—i; end;
```

kenapa cara ini kurang efisien? karna setiap elemen baru di tambahkan EMT harus menyalin seluruh isi v kembali ke variabel v.

Semakin efisien vektor telah ditentukan sebelumnya.

```
v=zeros(10000); for i=1 to 10000; v[i]=i; end;
```

Untuk mengubah tipe data di EMT, Anda dapat menggunakan fungsi seperti kompleks().

```
complex(1:4)
```

```
[ 1+0i , 2+0i , 3+0i , 4+0i ]
```

Konversi ke string hanya dimungkinkan untuk tipe data dasar. Format saat ini digunakan untuk penggabungan string sederhana. Tapi ada 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 jlength(v) then s=s+",";
endif; end; return s+""; endfunction j/prej * Variabel s diinisialisasi sebagai
string "" untuk menyimpan hasil * akhir. Awalnya, tanda kurung buka [
ditambahkan ke variabel s sebagai * pembuka.
```

```
* loop 1 to length(v); menjalankan perulangan dari elemen pertama * hingga
elemen terakhir dalam v. Fungsi length(v) mengembalikan panjang * atau jum-
lah elemen dalam vektor v.
```

```
* print(v[], 2, 0); adalah fungsi format yang mengonversi elemen * vektor v
pada posisi saat ini (v[]) menjadi string. * parameter 2 menunjukkan bahwa dua
digit setelah titik desimal akan * ditampilkan, sementara 0 memastikan bahwa
angka ditampilkan tanpa * tambahan simbol atau format lainnya.
```

```
* Bagian if lt;length(v) memeriksa apakah elemen saat ini bukan elemen *
terakhir. Jika benar, maka koma , akan ditambahkan ke variabel s untuk *
memisahkan elemen.
```

```
* Setelah loop selesai, tanda kurung tutup ] ditambahkan ke string s, * dan
string ini kemudian dikembalikan sebagai output.
```

```
tostr(linspace(0,1,10));
```

Untuk komunikasi dengan Maxima, terdapat fungsi convertmxm(), yang juga dapat digunakan untuk memformat vektor untuk keluaran.

```
convertmxm(1:10);
```

Untuk Latex perintah tex dapat digunakan untuk mendapatkan perintah Latex.

```
tex([1,2,3]);
```

[1, 2, 3]

## Faktor dan Tabel

Dalam pengantar R ada contoh yang disebut faktor.

Berikut ini adalah daftar wilayah 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"];
```

Perintah diatas digunakan untuk mendefinisikan sebuah array (array sendiri adalah sekumpulan variabel yang memiliki tipe data yang sama) karena pada

data tersebut ada beberapa nama negara bagian yang terulang. Array ini berisi singkatan untuk negara bagian dan teritori di Australia.

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 mari kita coba mencari nilai mean dan median dari data pendapatan tersebut menggunakan perintah `mean(incomes)` dan `median(incomes)`

```
mean(incomes)
54.7333333333
median(incomes)
57
```

Sekarang, kami ingin menghitung rata-rata pendapatan di suatu wilayah. Menjadi program statistik, R memiliki `faktor()` dan `tapply()` untuk ini.

EMT dapat melakukan hal ini dengan menemukan indeks wilayah dalam daftar wilayah unik.

```
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 itu, kita dapat menulis fungsi perulangan kita sendiri untuk melakukan sesuatu hanya untuk satu faktor.

Atau kita bisa 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 <
/pre > i : Parameter pertama biasanya adalah nilai yang digunakan untuk pencocokan atau pemetaan.
f: call : Parameter kedua, yang kemungkinan besar adalah sebuah fungsi yang dipanggil dalam kode tersebut.
```

di sini merujuk pada fungsi yang diterima sebagai input.

`cat`: Parameter ketiga adalah array atau vektor yang berisi kategori yang akan diproses.

`x`: Parameter keempat adalah array atau vektor yang akan diproses atau diubah berdasarkan pemetaan kategori yang dilakukan.

Ini agak tidak efisien, karena menghitung wilayah unik untuk setiap `i`, tetapi berhasil.

```
tappl(auterr,"mean",austates,incomes)
[44.5, 57.3333333333, 55.5, 53.6, 55, 60.5, 56, 52.25]
Perhatikan bahwa ini berfungsi untuk setiap vektor wilayah.
tappl(["act","nsw"],"mean",austates,incomes)
[44.5, 57.3333333333]
```

Sekarang, paket statistik EMT mendefinisikan tabel seperti di R. Fungsi `readtable()` dan `writetable()` dapat digunakan untuk input dan output.

Sehingga kita bisa mencetak rata-rata pendapatan negara di daerah secara bersahabat.

```
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
```

Fungsi `writetable` digunakan untuk menampilkan data dalam bentuk tabel yang terstruktur dengan label kolom dan lebar kolom yang dapat disesuaikan.

Dengan `labc=auterr`, berarti menetapkan label kolom untuk tabel tersebut berdasarkan kategori yang ada di `auterr` (yang sudah diurutkan sesuai abjad).



`wc(width of columns)=7` berarti setiap kolom dalam tabel akan memiliki lebar minimal 7 karakter.

sebagai contoh 44.5 itu memiliki 4 karakter (termasuk titik desimal).

karena data dalam kolom lebih pendek dari 7 karakter, kolom tersebut diberi ruang ekstra untuk tampilan yang rapi.

Kita juga bisa mencoba meniru perilaku R sepenuhnya.

Faktor-faktor tersebut harus disimpan dengan jelas dalam kumpulan beserta jenis dan kategorinya (negara bagian dan teritori dalam contoh kita). Untuk EMT, kami 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 i/prej statef=makef(austates);`

Perintah `statef = makef(austates);` digunakan untuk mengolah data yang ada dalam variabel `austates`, dan mengidentifikasi elemen unik yang ada dalam data tersebut.

Sekarang elemen ketiga dari koleksi akan berisi indeks.

```
statef[3]
```

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

`statef[3]` adalah elemen ketiga dari koleksi yang dikembalikan oleh fungsi `makef`, yaitu indeks posisi dari elemen-elemen dalam `austates` yang sudah dipetakan ke urutan dalam `u` (data unik yang terurut).

`statef[3]` akan mengembalikan indeks posisi dari setiap elemen dalam `austates` berdasarkan urutan yang ada di `u`.

Sekarang kita bisa meniru `tapply()` dengan cara berikut. Ini akan mengembalikan 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]); end i f; end; return x, uf; endfunction < /pre >
```

*Kamitidakmenambahkanbanyakpengecekantipedisini.Satu—satunyatindakanpencegahanmenyangkutkate*

Tabel ini dapat dicetak sebagai tabel dengan `writetable()`.

```
writetable(tapply(incomes,statef,"mean"),wc=7)
```

```
act nsw nt qld sa tas vic wa 44.5 57.33 55.5 53.6 55 60.5 56 52.25
```

Array

EMT hanya memiliki dua dimensi untuk array. Tipe datanya disebut matriks. Namun, akan mudah untuk menulis fungsi untuk dimensi yang lebih tinggi atau perpustakaan C untuk ini.

R memiliki lebih dari dua dimensi. Di R array adalah vektor dengan bidang dimensi.

Dalam EMT, vektor adalah matriks dengan satu baris. Itu dapat dibuat menjadi matriks dengan `redim()`.

```
shortformat; X=redim(1:20,4,5)
```

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

Fungsi `shortformat` digunakan untuk mengatur format tampilan angka agar lebih ringkas dan mudah dibaca.

Perintah diatas digunakan untuk membuat matrik X dari angka 1 sampai 20 dengan ketentuan matriks dengan 4 baris dan 5 kolom.

Ekstraksi baris dan kolom, atau sub-matriks, mirip dengan R.

```
X[,2:3]
```

```
2 3 7 8 12 13 17 18
```

Perintah diatas digunakan untuk menampilkan matriks X kolom kedua sampai ketiga.

```
X[,3:5]
```

```
3 4 5 8 9 10 13 14 15 18 19 20
```

Namun, di R dimungkinkan untuk menyetel daftar indeks vektor tertentu ke suatu nilai. Hal yang sama mungkin terjadi di EMT hanya dengan satu putaran.

```
function setmatrixvalue (M, i, j, v) ...
```

```
  loop 1 to max(length(i),length(j),length(v)) M[i,j] = v; end; endfunction
```

i/prej Perintah setmatrixvalue(M, i, j, v) adalah fungsi yang digunakan untuk mengubah nilai elemen-elemen dalam matriks berdasarkan indeks tertentu.

M: Matriks yang akan dimodifikasi.

i: Indeks baris atau posisi baris dalam matriks M yang ingin diubah.

j: Indeks kolom atau posisi kolom dalam matriks M yang ingin diubah.

v: Nilai yang akan dimasukkan ke dalam elemen-elemen matriks M pada posisi yang ditentukan oleh indeks i dan j.

Kami mendemonstrasikan ini untuk menunjukkan bahwa matriks dilewatkan dengan referensi di EMT. Jika Anda tidak ingin mengubah matriks M asli, Anda perlu menyalinnya ke 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
```

Perkalian luar dalam EMT hanya dapat dilakukan antar vektor. Ini otomatis karena bahasa matriks. Satu vektor harus berupa vektor kolom dan vektor lainnya harus berupa 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
```

1:5: Ini adalah vektor baris yang berisi angka-angka dari 1 hingga 5

(1:5)': Tanda ' di sini menunjukkan transposisi dari vektor baris 1:5. Dengan kata lain, ini mengubah vektor baris menjadi vektor kolom.

Dalam PDF pendahuluan untuk R terdapat contoh yang menghitung distribusi ab-cd untuk a,b,c,d yang dipilih dari 0 hingga n secara acak. Solusi dalam R adalah membentuk matriks 4 dimensi dan menjalankan table() di atasnya.

Tentu saja, hal ini dapat dicapai dengan satu putaran. Tapi loop tidak efektif di EMT atau R. Di EMT, kita bisa menulis loop di C dan itu akan menjadi solusi tercepat.

Namun kita ingin meniru perilaku R. Untuk melakukannya, kita perlu meratakan perkalian ab dan membuat matriks ab-cd.

```
a=0:6; b=a'; p=flatten(ab); q=flatten(p-p'); ... u=sort(unique(q)); f=getmultiplicities(u,q);
... statplot(u,f,"h");
```

```
![images/STATISTIKA_Lasigi
```

Selain multiplisitas eksak, EMT dapat menghitung frekuensi dalam vektor.

```
getfrequencies(q,-50:10:50)
[0, 23, 132, 316, 602, 801, 333, 141, 53, 0]
```

Perintah diatas digunakan untuk menghitung distribusi frekuensi nilai-nilai dalam vektor q dalam rentang dari -50 hingga 50, dengan interval 10. Fungsi ini menghitung berapa banyak nilai dalam q yang jatuh dalam setiap interval: [-50, -40), [-40, -30), ..., [40, 50).

Cara paling mudah untuk memplotnya sebagai distribusi adalah sebagai berikut.

```
plot2d(q,distribution=11):
!images/STATISTIKA_Lasigi
```

Namun dimungkinkan juga untuk menghitung terlebih dahulu penghitungan 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=" /"):
!images/STATISTIKA_Lasigi
```

Daftar

EMT memiliki dua jenis daftar. Salah satunya adalah daftar global yang bisa berubah, dan yang lainnya adalah tipe daftar yang tidak bisa diubah. Kami tidak peduli dengan daftar global di sini.

Tipe daftar yang tidak dapat diubah disebut koleksi di EMT. Ini berperilaku seperti struktur di C, tetapi elemennya hanya diberi nomor dan tidak diberi nama.

1. Membuat list dan mengakses elemen dalam list

```
L="Fred","Flintstone",40,[1990,1992]
Fred Flintstone 40 [1990, 1992]
```

Perintah diatas digunakan untuk membuat list L dengan nama depan Fred, nama belakang Flintstone, usia 40, dan tahun 1990, 1992.

Namun untuk tahun tersebut tidak dapat dipastikan apa arti dari tahun-tahun tersebut, bisa saja tahun kelahiran dan kematian, tahun pendidikan, tahun pekerjaan, atau yang lainnya.

Saat ini unsur-unsur tersebut tidak memiliki nama, meskipun nama dapat ditetapkan untuk tujuan khusus. Mereka diakses dengan nomor.

```
(L[4])[2]
1992
```

Perintah diatas digunakan untuk menampilkan list L keempat urutan kedua. Karena pada list L keempat berisi tahun yang dimana terdapat 2 tahun, tahun pertama adalah 1990 dan tahun kedua adalah 1992. Perintah tersebut ingin menampilkan tahun kedua, maka outputnya adalah 1992.

2. Menggabungkan dua list

```
A := [1,2,3]
[1, 2, 3]
B := [4,5,6]
[4, 5, 6]
C := [A, B]
```

```

[1, 2, 3, 4, 5, 6]
3. Mengubah elemen dalam list
D := [7,8,9,10]
[7, 8, 9, 10]
D[3] := 99
[7, 8, 99, 10]
4. menghitung panjang list
E := [10,20,30,40,50,60,70]
[10, 20, 30, 40, 50, 60, 70]
len := length(E)
7

```

File Input dan Output (Membaca dan Menulis Data)

Anda sering kali ingin mengimpor matriks data dari sumber lain ke EMT. Tutorial ini memberi tahu Anda tentang banyak cara untuk mencapai hal ini. Fungsi sederhananya adalah `writematrix()` dan `readmatrix()`.

Mari kita tunjukkan cara membaca dan menulis vektor real ke file.

```

a=random(1,100); mean(a), dev(a),
0.40761 0.295

```

Untuk menulis data ke file, kita menggunakan fungsi `writematrix()`.

Karena pengenalan ini kemungkinan besar ada di direktori, di mana pengguna tidak memiliki akses tulis, kami menulis data ke direktori home pengguna. Untuk buku catatan sendiri, hal ini tidak diperlukan, karena file data akan ditulis ke dalam direktori yang sama.

```

filename="test.dat";

```

Sekarang kita menulis vektor kolom `a'` ke file. Ini menghasilkan satu nomor di setiap baris file.

```

writematrix(a',filename)

```

Untuk membaca data, kita menggunakan `readmatrix()`

```

a=readmatrix(filename)'

```

```

[0.83054, 0.99006, 0.064585, 0.57221, 0.16539, 0.31818, 0.16037, 0.71343,
0.6188, 0.67365, 0.22367, 0.86965, 0.67439, 0.859, 0.85639, 0.07405, 0.65051,
0.79298, 0.039088, 0.4536, 0.24248, 0.10425, 0.36493, 0.084869, 0.37012, 0.045487,
0.66347, 0.69294, 0.46928, 0.27625, 0.20081, 0.069966, 0.2009, 0.52929, 0.97226,
0.52319, 0.11857, 0.075214, 0.022133, 0.24327, 0.031857, 0.96373, 0.12786, 0.19148,
0.93015, 0.95501, 0.5936, 0.48596, 0.14792, 0.23486, 0.086331, 0.012303, 0.34125,
0.58942, 0.15009, 0.49116, 0.28172, 0.35863, 0.1159, 0.2136, 0.34576, 0.32714,
0.55376, 0.67384, 0.042919, 0.2402, 0.086114, 0.36787, 0.13432, 0.36163, 0.33836,
0.26447, 0.96735, 0.31812, 0.5575, 0.32324, 0.44814, 0.31933, 0.59803, 0.69466,
0.86443, 0.2422, 0.43268, 0.78779, 0.26504, 0.034781, 0.096976, 0.54639, 0.43574,
0.98703, 0.071198, 0.40604, 0.9838, 0.90533, 0.82669, 0.30697, 0.076151, 0.15164,
0.18336, 0.020708]

```

Dan hapus file tersebut.

```

fileremove(filename);

```

```

mean(a), dev(a),

```

```

0.40761 0.295

```

Fungsi `writematrix()` atau `writetable()` dapat dikonfigurasi untuk bahasa lain.

Misalnya, jika Anda memiliki sistem Indonesia (titik desimal dengan koma), Excel Anda memerlukan nilai dengan koma desimal yang dipisahkan dengan titik koma dalam file csv (defaultnya adalah nilai yang dipisahkan koma). File berikut "test.csv" akan muncul di folder saat ini Anda.

```
filename="test.csv"; ... writematrix(random(5,3),file=filename,separator=",")
```

Anda sekarang dapat membuka file ini dengan Excel bahasa Indonesia secara langsung.

```
fileremove(filename);
```

Terkadang kita memiliki string dengan token seperti berikut.

```
s1:="f m m f m m m f f f m m f"; ... s2:="f f f m m f f";
```

Untuk melakukan tokenisasi ini, kami 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 header 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();
```

The file looks like this.

```
printfile(file)
```

```
A,B,C 0.2057919970212131,0.3673199858619826,0.1235850721154041 0.2729556337935984,0.9103935604837
0.1223544186913062,0.6908957011653737,0.621387244117407
```

Fungsi `readtable()` dalam bentuknya yang paling sederhana dapat membaca ini dan mengembalikan kumpulan nilai dan baris judul.

```
L=readtable(file, list);
```

Koleksi ini dapat dicetak dengan `writetable()` ke buku catatan, atau ke file.

```
writetable(L,wc=10,dc=5)
```

```
A B C 0.20579 0.36732 0.12359 0.27296 0.91039 0.39342 0.12235 0.6909
0.62139
```

Matriks nilai adalah elemen pertama dari L. Perhatikan bahwa `mean()` di EMT menghitung nilai rata-rata baris matriks.

```
mean(L[1])
```

```
0.23223 0.52559 0.47821
```

File CSV

Pertama, mari kita menulis matriks ke dalam file. Untuk outputnya, kami membuat file di direktori kerja saat ini.

```
file="test.csv"; ... M=random(3,3); writematrix(M,file);
```

Here is the content of this file.

```
printfile(file)
```

```
0.1584878123698123,0.3255074907411526,0.498753624686067 0.4497303404971393,0.008885263351373088,0.05045626731067091,0.6970445235487002,0.1027538621491186
```

CVS ini dapat dibuka pada sistem berbahasa Inggris ke Excel dengan klik dua kali. Jika Anda mendapatkan file seperti itu di sistem Jerman, Anda perlu mengimpor data ke Excel dengan memperhatikan titik desimal.

Namun titik desimal juga merupakan format default untuk EMT. Anda dapat membaca matriks dari file dengan `readmatrix()`.

```
readmatrix(file)
```

```
0.15849 0.32551 0.49875 0.44973 0.0088853 0.99598 0.50456 0.69704 0.10275
```

Dimungkinkan untuk menulis beberapa matriks ke satu file. Perintah `open()` dapat membuka file untuk ditulis dengan parameter "w". Standarnya adalah "r" untuk membaca.

```
open(file,"w"); writematrix(M); writematrix(M'); close();
```

Matriks dipisahkan oleh garis 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 koma). Di Excel 2007, gunakan "save as" dan "other format", lalu pilih "CSV". Pastikan tabel saat ini hanya berisi data yang ingin Anda ekspor.

Ini sebuah contoh.

```
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 bahasa Jerman saya menggunakan titik koma sebagai pemisah dan koma desimal. Anda dapat mengubahnya di pengaturan sistem atau di Excel, tetapi hal ini tidak diperlukan untuk membaca matriks menjadi EMT.

Cara termudah untuk membaca ini ke dalam Euler adalah `readmatrix()`. Semua koma diganti dengan titik dengan parameter `gt;koma`. 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]');
```

```
!images/STATISTIKALasigi
```

Ada cara yang lebih mendasar untuk membaca data dari suatu file. Anda dapat membuka file dan membaca angka baris demi baris. Fungsi `getvectorline()` akan membaca angka dari sebaris data. Secara default, ini mengharapkan titik desimal. Tapi bisa juga menggunakan koma desimal, jika Anda memanggil `setdecimaldot(",")` sebelum Anda menggunakan fungsi ini.

Fungsi berikut adalah contohnya. Itu akan berhenti di akhir file atau baris kosong.

```
function myload (file) ...
    open(file); M=[]; repeat until eof(); v=getvectorline(3); if length(v)~0 then
M=M_v; else break; end; end; return M; close(file); endfunction < /pre > myload(file)
0.15849 0.32551 0.49875 0.44973 0.0088853 0.99598 0.50456 0.69704 0.10275
```

Dimungkinkan juga untuk membaca semua angka dalam file itu dengan `getvector()`.

```
open(file); v=getvector(10000); close(); redim(v[1:9],3,3)
0.15849 0.32551 0.49875 0.44973 0.0088853 0.99598 0.50456 0.69704 0.10275
```

Oleh karena itu sangat mudah untuk menyimpan suatu vektor nilai, satu nilai di setiap baris dan membaca kembali vektor ini.

```
v=random(1000); mean(v)
0.49078
writematrix(v',file); mean(readmatrix(file)')
0.49078
```

Menggunakan Tabel

Tabel dapat digunakan untuk membaca atau menulis data numerik. Misalnya, kita menulis tabel dengan header baris dan kolom ke sebuah file.

```
file="test.tab"; M=random(3,3); ... open(file,"w"); ... writetable(M,separator=",",labc=["one","two"],"
... close(); ... printfile(file)
```

```
one,two,three 0.94, 0.31, 0.36 0.36, 0.79, 0.88 0.45, 0.29, 0.82
```

Ini dapat diimpor ke Excel.

Untuk membaca file di EMT, kami menggunakan `readtable()`.

```
M,headings=readtable(file, clabs); ... writetable(M,labc=headings)
```

```
one two three 0.94 0.31 0.36 0.36 0.79 0.88 0.45 0.29 0.82
```

Menganalisis Garis

Pada subbab ini sering digunakan untuk memproses atau mengekstrak data dari teks yang berformat khusus, seperti data tabel dalam HTML. 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 akan memisahkan string line menjadi bagian-bagian yang lebih kecil, yang dikenal sebagai "token".

```
vt=strtokens(line)
```

```
2020-11-03 Tue 1'114.05
```

Kemudian kita dapat mengevaluasi setiap elemen garis menggunakan evaluasi yang sesuai.

```
day(vt[1]); ... indexof(["mon","tue","wed","thu","fri","sat","sun"],tolower(vt[2]));
... strrepl(vt[3],"","")();
```

Dengan menggunakan ekspresi reguler, dimungkinkan untuk mengekstrak hampir semua informasi dari sebaris data.

Selanjutnya, kita akan melihat bagaimana mengekstrak data string yang berisi markup HTML menggunakan ekspresi reguler.

```
line="|tr |td 1145.45|td 5.6|td -4.5|td |tr ";
```

Untuk mengekstraknya, kami menggunakan ekspresi reguler, yang mencari

- tanda kurung tutup gt;, untuk mengindikasikan bahwa kita akan mencari awal dari elemen yang ada di dalam tag.
- string apa pun yang tidak mengandung tanda kurung akan mencocokkan elemen di dalam tag lt;tdgt;.
- braket pembuka dan penutup menggunakan solusi terpendek, dengan tag pembuka (lt;tdgt;) dan penutup (lt;/tdgt;).
- sekali lagi string apa pun yang tidak mengandung tanda kurung, ini akan menjamin bahwa kita akan mengambil isi yang relevan di dalam tagnya.
- dan tanda kurung buka lt; menandai bahwa ini adalah akhir dari tag dan awal dari tag baru.

Mencari pola tertentu dalam string line yang menggunakan ekspresi reguler.  
 pos,s,vt=strxfind(line,"([< ]+)<.+?([< ]+)<");

Hasilnya adalah posisi kecocokan, string yang cocok, dan vektor string untuk sub-kecocokan.

Kita akan mengeksekusi elemen-elemen di dalam array atau list vt satu per satu dalam sebuah perulangan.

```
for k=1:length(vt); vt[k](), end;
```

1145.5 5.6

Berikut adalah fungsi yang membaca semua item numerik antara lt;tdgt; dan lt;/tdgt;.

```
function readtd (line) ...
```

```
v=[]; cp=0; repeat pos,s,vt=strxfind(line,"jtd.*?;(.+?)i/td;",cp); until pos==0;
if length(vt)>0 then v=v—vt[1]; endif; cp=pos+strlen(s); end; return v; end-
function j/pre; Kita akan mengekstrak dan menampilkan semua nilai yang be-
rada di antara tag lt;tdgt;...lt;/tdgt; dalam baris, dan mencari apakah nilai terse-
but numerik atau bukan.
```

```
readtd(line+"jtd non-numericali/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 terkini dari situs EMT. Kami menggu-  
 nakan ekspresi reguler untuk memindai "Versi ..." dalam sebuah judul.

```
function readversion () ...
```

```
urlopen("http://www.euler-math-toolbox.de/Programs/Changes.html"); re-
peat until urlof(); s=urlgetline(); k=strfind(s,"Version ",1); if k>0 then substring(s,k,strfind(s,"i",k)-
1), break; endif; end; urlclose(); endfunction j/pre; readversion
```

Version 2024-01-12

Contoh lain membaca URL dengan EMT

```
"https://mywebsite.com/version.h"
```

```
function readversionmywebsite () ...
```

```
urlopen("https://mywebsite.com/version.h"); repeat until urlof(); s=urlgetline();
k=strfind(s,"Release",1); if k>0 then substring(s,k,strfind(s,"i",k)-1); break; en-
dif; end; urlclose(); endfunction j/pre; readversionmywebsite
```



Karena string "Release" tidak ada di dalam file version.h, maka strfind(s, "Release", 1) akan mengembalikan nilai nol atau tidak menghasilkan indeks yang diperlukan untuk proses pencarian.

Input dan Output 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="tes.e"; ... writevar(random(2,2),"M",file); ... printfile(file,3)
```

```
M = [ .. 0.4872074455081638, 0.8740865049405735; 0.5613621150682586,
0.004109765099980884];
```

Sekarang kita dapat memuat file tersebut. Ini akan mendefinisikan matriks M.

```
load(file); show M,
M = 0.48721 0.87409 0.56136 0.0041098
```

Omong-omong, jika writevar() digunakan pada suatu variabel, definisi variabel dengan nama variabel tersebut akan dicetak.

```
writevar(M); writevar(inch)
```

```
M = [ .. 0.4872074455081638, 0.8740865049405735; 0.5613621150682586,
0.004109765099980884]; inch= 0.0254;
```

Kita juga bisa membuka file baru atau menambahkan file yang sudah ada.

Dalam contoh kita menambahkan file yang dibuat sebelumnya.

```
open(file,"a"); ... writevar(random(2,2),"M1"); ... writevar(random(3,1),"M2");
... close();
load(file); show M1; show M2;
```

```
M1 = 0.03006 0.2823 0.24361 0.31622 M2 = 0.63099 0.1284 0.1631
```

Untuk menghapus file apa pun, gunakan fileremove().

```
fileremove(file);
```

Vektor baris dalam suatu file tidak memerlukan koma, jika setiap angka berada pada baris baru. Mari kita buat file seperti itu, tulis 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 = [ 0.496372610885 0.728479886391 0.560673240413 0.258032232026 0.306613101653
];
```

```
load(file); M
[0.49637, 0.72848, 0.56067, 0.25803, 0.30661]
```

#### LATIHAN

1. Misalkan anda memiliki vektor x=[2,4,6,8,10]

a. buatkan vektor yang menggabungkan vektor x,angka0dan vektorx lagi

b. tentukan apakah setiap elemen vektor x lebih besar dari 5(hasil logika 1 untuk benar dan 0 untuk salah)

```
x:=[2,4,6,8,10]; [x,0,x]
```

```
[2, 4, 6, 8, 10, 0, 2, 4, 6, 8, 10]
```

```
x 5,
```

```

[0, 0, 1, 1, 1] [0, 0, 6, 8, 10]
2. Tentukan matriks X dengan elemen-elemen yang berurutan dari 1 hingga
20 dan susunlah elemen tersebut menjadi matriks berukuran 5x4.
shortformat; X=redim(1:20,5,4)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
3. Seorang analis memiliki data penjualan harian selama 5 hari(150,200,250,300,350)
yang disimpan dalam bentuk vektor sebagai berikut:
a. mean(rata-rata)
b. deviasi standar
penjualan=[150,200,250,300,350]
[150, 200, 250, 300, 350]
atau anda bisa memanggil data yang sudah dibuat
filename="penjualan.dat";
writematrix(penjualan,filename)
penjualan=readmatrix(filename)'
[150, 200, 250, 300, 350]
mean(penjualan)
250
dev(penjualan)
79.057
4. Buat fungsi yang membuka URL
"https://en.wikipedia.org/wiki/Euler(software)"
dan mencari kata "Versi" di dalam URL tersebut, dan tampilkan hasilnya.
function readversionwebsite () ...
urlopen("https://en.wikipedia.org/wiki/Euler(software)");repeatuntilurleof(); s =
urlgetline(); k = strfind(s, "version", 1); if k > 0 then substring(s, k, strfind(s, " <
", k) - 1), break; end; if; end; urlclose(); endfunction < /pre > readversion
Version 2024-01-12
5. Diberikan data pengukuran tinggi badan pada kelas matematika B adalah
sebagai berikut:
— Rentang Tinggi (cm) — Jumlah Orang — —————
— — 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 —
a.) Hitung rata-rata dan deviasi standar dari distribusi tinggi badan ini.
b.) Plot distribusi frekuensi data (diagram batang).
c.) Tambahkan kurva distribusi normal untuk dibandingkan dengan data.
r = 155.5:4:187.5 // Rentang ukuran tinggi badan
[155.5, 159.5, 163.5, 167.5, 171.5, 175.5, 179.5, 183.5, 187.5]
v = [22, 71, 136, 169, 139, 71, 32, 8] // Jumlah orang dalam tiap rentang
[22, 71, 136, 169, 139, 71, 32, 8]
l=fold(r,[0.5,0.5]) // Menghitung titik tengah dari setiap rentang tinggi badan
[157.5, 161.5, 165.5, 169.5, 173.5, 177.5, 181.5, 185.5]
m,d=meandev(l,v); m, d, // Hitung rata-rata dan deviasi standar
169.9 5.9891

```

```

    plot2d(r, v, a=150, b=200, c=0, d=190, bar=1, style="
/")):
    ![images/STATISTIKALasigi
    plot2d("qnormal(x, m, d) sum(v) 4", ... xmin=min(r), xmax=max(r),
thickness=3, add=1):
    ![images/STATISTIKALasigi
    remvalue();
6. Sebuah survei dilakukan untuk mengetahui jumlah jam belajar siswa
SMA dalam satu minggu. Berikut data jam belajar dari 10 siswa: 8, 10, 7, 6,
9, 10, 11, 9, 8, 12.
a) Hitung nilai rata-rata dari data di atas
b) Tentukan median dari data tersebut.
M=[8,10,7,6,9,10,11,9,8,12];
mean(M)
9
median(M)
9
7. Anda diberikan data yang menunjukkan jumlah penjualan barang selama
12 bulan dalam satu tahun berturut-turut 120, 135, 150, 160, 170, 180, 190,
210, 200, 220, 230, 240.
a) Buatlah plot garis dari data penjualan barang tersebut.
b) Hitung rata-rata penjualan perbulan.
X=[120,135,150,160,170,180,190,210,200,220,230,240]
[120, 135, 150, 160, 170, 180, 190, 210, 200, 220, 230, 240]
Y=[1,2,3,4,5,6,7,8,9,10,11,12]
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
statplot(Y,X,"l"):
![images/STATISTIKALasigi
mean(X)
183.75

```