

# Tugas Akhir Aplikom

Marcelline Calya Padmarini

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. \* — \* 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 *"ntukeksponen(pangkat)*.

\* Seperti biasa, \* dan / bersifat lebih kuat daripada + atau -.

\* *"mengikat lebih kuat dari \*"*, *sehinggapi\*r<sup>2</sup> merupakan rumus luas\*lingkaran*.

\* Jika perlu, Anda harus menambahkan tanda kurung, seperti pada 2 (2<sup>3</sup>).

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

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

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

`1.25`

Argumen atau input untuk fungsi ditulis di dalam tanda kurung.

`sin(45°), cos(pi), log(sqrt(E))`

`0.707106781187 -1 0.5`

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

EMT menyediakan banyak sekali fungsi dan operator matematika. Hampir semua fungsi matematika sudah tersedia di EMT. Anda dapat melihat daftar lengkap fungsi-fungsi matematika di EMT pada berkas Referensi (klik menu `Help -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. \* —

Satuan

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

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

`1609.344`

Beberapa satuan yang sudah dikenal di dalam EMT adalah sebagai berikut.

Semua unit diakhiri dengan tanda dolar (`)`, *namun boleh tidak perlu ditulis dengan mengaktifkan 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 sebelumnya. Untuk menggabungkan suatu baris ke baris sebelumnya, gunakan [Ctrl]+[Backspace].

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

```

a=4; b=15; c=2; // menyelesaikan  $ax^2 + bx + c = 0$  secara manual... 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(x^2 + 5x + 6)
2 2 b + 2 a b + a
(x + 2) (x + 3)
solve(ax^2 + bx + c, x)//rumusabc
2 2 (- sqrt(b - 4 a c)) - b sqrt(b - 4 a c) - b [x = -----, x =
-----] 2 a 2 a
(a^2 - b^2)/(a + b), ratsimp(
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 "amp;=".

```

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 // (33 + 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 perhitungagn simbolik secara lebih bagus menggunakan LaTeX. Untuk melakukan hal ini, tambahkan tanda dolar (\$) *didepantanda&padasetiapperintah*

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

```
(a + b)2
expand((a + b)2),factor(x2 + 5x + 6)
solve(ax2 + bx + c, x)//rumusabc
(a2 - b2)/(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.

LATIHAN EMT MARCELLINE

Latihan 1 Perintah Baru

String

Sebuah string dalam Euler didefinisikan dengan "..."

Contoh

"String dapat berisikan apapun"

String dapat berisikan apapun

String dapat digabungkan dengan — atau + serta dapat juga berfungsi dengan angka yang mampu diubah menjadi sebuah string

"Luas persegi dengan panjang sisi " + 5 + " adalah " + 5<sup>2</sup> + "cm<sup>2</sup>"

Luas persegi dengan panjang sisi 5 adalah 25 cm<sup>2</sup>

atau dapat ditulis dengan

"Luas persegi dengan panjang sisi " —5— " adalah " —55— " cm<sup>2</sup>"

Luas persegi dengan panjang sisi 5 adalah 25 cm<sup>2</sup>

Nilai boolean

Nilai boolean ini direpresentasikan dengan 1 adalah benar atau 0 adalah salah. String dapat dibandingkan, sama seperti angka

Contoh

20 12, "jeruk" "melon"

1 0

Terdapat fungsi "nonzeros (mana yang merupakan fungsi benar pada syarat sebelumnya

(1:15);10, nonzeros(

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

Karena angka yang diambil ialah 1-15, maka nilai yang kurang dari 10 adalah 1-9.

---

Latihan 2 Perintah Baru dan Contoh Perhitungan

Matriks

Sebuah matriks dalam EMT dimasukkan dengan menggunakan tanda kurung siku "...". Nilai setiap baris matriks tersebut dipisahkan dengan koma, dan setiap kolom dipisahkan dengan titik koma

a = 1; b = 2; c = 3; d = 4; [a,b;c,d]

1 2 3 4

t:= 3;

short [t,1;0,t,t;t,0,1,t]

3 1 0 0 0 3 3 0 3 0 1 3

dengan syarat yang ada diatas yaitu

a = 1, b = 2, c = 3, d = 4

sebuah matriks dapat pula ditulis menjadi

matrix([a,b],[c,d])

1 2 3 4

zeros(3,3)

0 0 0 0 0 0 0 0 0

Untuk membuat matriks identitas dapat dilakukan dengan

id(3)

1 0 0 0 1 0 0 0 1

CONTOH SOAL:

diberikan matriks A = [2,3],[1,5] dan matriks B = [1,4],[2,5]. Maka tentukan nilai 3A+2B, A\*A, B\*B, A\*B!

A:=[2,3;1,5]

2 3 1 5

B:=[1,4;2,5]

1 4 2 5

3A+2B

8 17 7 25

AA

4 9 1 25

BB

1 16 4 25

AA

4 9 1 25

AB

2 12 2 25

dari penyelesaian tersebut didapat nilai dari yang akan dicari

—

Latihan 3

Soal-Soal

1. Tentukan nilai  $x_1$  dan  $x_2$  dari  $3x^2 - 5x = -1$ !

Jawab:

Apabila penyelesaian yang dilakukan menggunakan LaTeX, maka harus diawali dengan tanda "amp;"

A:= solve(3(x^2) - 5x + 1, x)//menggunakan rumus abc

$$\left[ x = \frac{5 - \sqrt{13}}{6}, x = \frac{\sqrt{13} + 5}{6} \right]$$

Sedangkan, apabila penyelesaian dilakukan dengan perhitungan matematika simbolis diawali dengan tanda "amp;"

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

5 - sqrt(13) sqrt(13) + 5 [x = ———, x = ———] 6 6

kedua cara tersebut akan menghasilkan nilai x yang sama, sehingga kedua cara dapat dilakukan.

Namun pastikan jika ingin menggunakan cara 1 sudah menginstall LaTeX

2. Tentukan nilai  $x_1$  dan  $x_2$  dari  $2(x^2) + 3x - 5$  dan gambarkan grafiknya!

Jawab:

function map f(x):= (2(x^2) + 3x - 5)

solve(2(x^2) + 3x - 5, x)

5 [x = 1, x = -] 2

plot2d(2(x^2) + 3x - 5, 1, -(5/2))

—

3. Diberikan  $f(x) = x^3 - 6x^2 - 2x + 40$ , hitung nilai  $f(-5)$  dan  $f(4)$ !

Jawab:

Sebelum menghitung nilai, kita perlu menuliskan ulang nilai  $f(x)$ . Menjadi

function f(x):= (x^3) - 6(x^2) - 2x + 40

Akan diselesaikan dengan cara yang sederhana yaitu

f(-5)

-225

f(4)

0

## 2 Aljabar

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

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

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

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

$$-\frac{42}{xy^4}$$

Menjabarkan:

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

$$\text{showev}(\text{'expand}((6x(-3) + y^5)(-7x^2 - y(-9))))$$

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

Baris Perintah

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

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

r:=2; h:=4; pi\*r<sup>2</sup>\*h/3

16.7551608191

Perintah harus dipisahkan dengan spasi kosong. Baris perintah berikut mencetak kedua hasilnya.

pi\*2rh,

50.2654824574 100.530964915

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

x := 1;

```

x := cos(x) // nilai cosinus (x dalam radian)
0.540302305868
x := cos(x)
0.857553215846

```

Jika 2 baris dihubungkan dengan "..." kedua baris tersebut akan selalu dieksekusi secara bersamaan.

```

x := 1.5; ... x := (x+2/x)/2, x := (x+2/x)/2, x := (x+2/x)/2,
1.41666666667 1.41421568627 1.41421356237

```

Ini adalah jalan terbaik untuk menyebarkan perintah panjang ke-2 atau lebih baris. Anda dapat menekan Ctrl+Return untuk memisahkan baris menjadi 2 pada posisi kursor saat ini, atau Ctrl+Back untuk menggabungkan baris.

Untuk melipat semua multi-baris tekan Ctrl+L. Kemudian baris-baris berikutnya hanya akan terlihat jika salah satunya memiliki fokus. Untuk melipat 1 multi-baris, mulai baris pertama dengan "

```

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

```

Euler mendukung perulangan di baris perintah, selama mereka muat dalam 1 baris tunggal atau multi-baris. Dalam program, batasan ini tidak berlaku, tentu saja. Untuk informasi lebih lanjut, lihat pengantar berikut.

```

x=1; for i=1 to 5; x := (x+2/x)/2, end; // menghitung akar 2
1.5 1.41666666667 1.41421568627 1.41421356237 1.41421356237

```

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

```

x := 1.5; // comments go here before the ... repeat xnew:=(x+2/x)/2;
until xnew =x; ... x := xnew; ... end; ... x,
1.41421356237

```

```

Struktur kondisional juga berfungsi
if  $E^p i \pi^E$ ; then "Thoughtso!", endif;
Thought so!

```

Ketika Anda mengeksekusi perintah, kursor dapat berada di posisi mana saja dalam 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 pergi ke perintah.

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

```

sqrt(sin(10°)/cos(20°))
0.429875017772

```

Untuk melihat bantuan untuk perintah terakhir, buka jendela bantuan dengan F1. Disana anda dapat memasukan teks untuk dicari. Pada baris kosong, bantuan untuk jendela bantuan akan ditampilkan. Anda dapat menekan escape (esc) untuk menghapus baris, atau untuk menutup jendela bantuan.

Anda dapat klik 2 kali di perintah apapun untuk membuka bantuan pada perintah ini. Coba klik 2 kali perintah exp dalam baris perintah.

```

exp(log(2.5))

```

## 2.5

Anda juga dapat menyalin dan menempel di Euler. Menggunakan Ctrl+C dan Ctrl+v untuk ini. Untuk menandai teks, seret mouse atau memakai shift bersamaan dengan tombol kursor apapun. Selain itu, anda dapat menyalin kurung yang disorot.

### Sintaks Dasar

Euler mengetahui fungsi matematika yang biasa digunakan. Seperti anda lihat sebelumnya, fungsi trigonometri bekerja di radian atau derajat. Untuk menkonversi ke derajat, tambahkan simbol derajat (dengan tombol F7) ke nilainya, atau menggunakan fungsi rad (x). Fungsi akar kuadrat disebut sqrt di Euler. Tentu saja,  $x^{(1/2)}$  *jugadimungkinkan*.

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

Beberapa perintah dalam 1 baris dipisahkan dengan ";", atau ";". Titik koma menekan keluaran perintah. Di akhir baris perintah, ";", diasumsikan jika ";", hilang.

```
g:=9.81; t:=2.5; 1/2gt^2
30.65625
```

EMT menggunakan sintaks program untuk ekspresi. Untuk memasukan

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

anda harus mengatur tanda kurung yang benar dan gunakan / untuk pecahan. Perhatikan tanda kurung yang disorot untuk bantuan. Perhatikan bahwa kontanta Euler e dinamakan E di EMT.

```
E^2(1/(3 + 4log(0.6)) + 1/7)
8.77908249441
```

Untuk menghitung ekspresi yang rumit seperti

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

anda perlu memasukannya dalam bentuk baris.

```
((1/7 + 1/8 + 2) / (1/3 + 1/2))^2pi
23.2671801626
```

Letakan kurung dengan hati-hati disekitar sub-ekspresi yang perlu dihitung terlebih dahulu. EMT membantu anda untuk menyorot ekspresi yang diakhiri oleh kurung tutup. Anda juga harus memasukan nama "pi" untuk huruf p Yunani.

Hasil dari perhitungan ini adalah bilangan titik pengambang. Secara default, ini dicetak dengan akurasi sekitar 12 digit. Dalam baris perintah berikut, kami juga mempelajari cara merujuk ke hasil sebelumnya dalam baris yang sama.

```
1/3+1/7, fraction
0.47619047619 10/21
```



Fungsi print juga mampu mengkonversikan angka menjadi sebuah kalimat. Fungsi ini dapat menerima sejumlah digit dan sejumlah tempat (0 untuk keluaran padat), dan secara optimal sebuah unit.

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

```
Golden Ratio : 1.61803
```

Terdapat string khusus none yang tidak dicetak. Ini dikembalikan oleh beberapa fungsi, ketika hasilnya tidak penting. (Ini dikembalikan secara otomatis, jika fungsi tidak memiliki pernyataan return).

```
none
```

Untuk mengkonversi sebuah kalimat menjadi angka, cukup evaluasi saja. Ini juga berfungsi untuk ekspresi (lihat dibawah).

```
"1234.5"()
```

```
1234.5
```

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

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

```
affe charlie bravo
```

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

```
w:["none"]; w—v—v
```

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

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

String Unicode dapat dikonkatenasikan seperti string lainnya.

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

```
I
```

Dalam komentar, entitas yang sama seperti , etc. dapat digunakan. Ini mungkin merupakan alternatif cepat untuk lateks. (detail lebih lanjut tentang komentar dibawah).

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

```
v=strttochar(u"Auml; is a German letter")
```

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

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

```
v[1]=strttochar(u"Uuml;")[1]; chartoutf(v)
```

```
Ü is a German letter
```

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

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

```
We have =.
```

Ini juga mungkin untuk menggunakan entitas numerik.



```

u"196;hnliches"
Ähnliches
Nilai Boolean
Nilai boolean direpresentasikan dengan 1 =benar atau 0=salah dalam Euler.String dapat dibandingkan sama dengan halnya angka.
2;1, "apel";"banana"
0 1
"and" adalah operator "&&"; dan "or" merupakan operator "——", seperti dalam bahasa C. (Kata-kata "and" dan "or" hanya dapat digunakan dalam kondisi "if").
2;E E;3
1
Operasi boolean mematuhi aturan bahasa matriks.
(1:10) 5, nonzeros(
[0, 0, 0, 0, 0, 1, 1, 1, 1, 1] [6, 7, 8, 9, 10]
Anda dapat menggunakan fungsi nonzeros() untuk mengekstrak elemen spesifik dari vektor. Seperti contoh, kami menggunakan kondisi isprime(n).
N=2—3:2:99 // N berisi elemen 2 dan bilangan2 ganjil dari 3 s.d. 99
[2, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99]
N[nonzeros(isprime(N))] //pilih anggota2 N yang prima
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97]
Format Keluaran
Format keluaran default dari EMT adalah mencetak 12 digit. Untuk memastikan bahwa kita melihat defaultnya, kami mereset formatnya.
defformat; pi
3.14159265359
Secara internal, EMT menggunakan standar IEEE untuk bilangan double dengan sekitar 16 digit desimal. Untuk melihat jumlah digit penuh, gunakan perintah "longestformat", atau kita gunakan operator "longest" untuk menampilkan hasilnya dalam format terpanjang.
longest pi
3.141592653589793
Berikut adalah representasi heksadesimal internal dari bilangan double.
printhe(pi)
3.243F6A8885A30*160
Format keluaran dapat diubah secara permanen dengan perintah format.
format(12,5); 1/3, pi, sin(1)
0.33333 3.14159 0.84147
Defaultnya adalah format(12).
format(12); 1/3
0.333333333333
Fungsi seperti "shortesformat", "shortformat", "longformat" bekerja untuk vektor yang diikuti.

```

```

shortestformat; random(3,8)
0.6 0.59 0.032 0.053 0.6 0.56 0.84 0.18 0.4 0.84 0.026 0.66 0.63 0.77 0.67 0.83
0.98 0.54 0.18 0.7 0.71 0.93 0.49 0.02

```

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

```

setscalarformat(5); pi
3.1416

```

Fungsi "longestformat" juga mengatur format skalar.

```

longestformat; pi
3.141592653589793

```

Sebagai referensi, berikut adalah daftar format keluaran yang paling penting

```

shortestformat shortformat longformat, longestformat format(length,digits)
goodformat(length) fracformat(length) defformat

```

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

Tetapi format keluaran EMT dapat diatur secara fleksibel.

```

longestformat; pi,
3.141592653589793
format(10,5); pi
3.14159

```

Defaultnya adalah defformat().

```

defformat; // default

```

Terdapat operator pendek yang hanya mencetak 1 nilai. Operator "longest" akan mencetak semua digit valid dari suatu angka.

```

longest pi^2/2
4.934802200544679

```

Ada juga operator pendek untuk mencetak hasil dalam format pecahan.

Kami telah menggunakannya diatas.

```

fraction 1+1/2+1/3+1/4
25/12

```

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

```

longest 0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1-1
-1.110223024625157e-16

```

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

```

0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1-1
0

```

Ekspresi

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

Variabel global dapat digunakan dalam evaluasi.

```

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

```

12.56637061435917

Parameter ditetapkan ke x, y, dan z dalam urutan itu. Parameter tambahan dapat ditambah dengan menggunakan parameter yang telah ditetapkan.

```
fx:="asin(x)2"; fx(5, a = -1)
-0.919535764538
```

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

```
at:=4; function f(expr,x,at) := expr(x); ... f("atx2", 3, 5)//computes 4 × 32 not 53
```

36

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

```
at:=4; function f(expr,x,a) := expr(x,at=a); ... f("atx2", 3, 5)
```

45

Sebagai referensi, kami menyatakan bahwa koleksi panggilan (didiskusikan di tempat lain) dapat berisi ekspresi. Sehingga kita dapat membuat contoh di atas sebagai berikut.

```
at:=4; function f(expr,x) := expr(x); ... f("atx2", at = 5, 3)
```

45

Ekspresi dalam x sering digunakan seperti fungsi.

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

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

12

Sebagai konvensi, ekspresi simbolik atau numerik harus dinamai dengan fx, fxy, dsb. Skema penamaan ini tidak boleh digunakan untuk fungsi.

```
fx = diff(xx, x); fx
```

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

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

```
"@(a,b) a2 + b2",
@(a,b) a2 + b2
```

41

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

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

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

```
fx = x3 - ax; ... a = 1.2; fx(0.5)
-0.475
```

Untuk semua variabel lain dalam ekspresi dapat dispesifikasikan dalam evaluasi menggunakan parameter yang ditetapkan.

```
fx(0.5,a=1.1)
-0.425
```

Sebuah ekspresi tidak perlu bersifat simbolik. Hal ini diperlukan, jika ekspresi mengandung sebuah fungsi, yang hanya diketahui dalam kernel numerik, bukan di Maxima.

Simbol Matematika

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

Simbol matematika diintegrasikan secara mulus ke dalam Euler dengan amp;. Setiap ekspresi yang dimulai dengan amp; adalah ekspresi simbolik. Ini dievaluasi dan dicetak oleh Maxima.

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

```
44!
```

```
2658271574788448768043625811014615890319638528000000000
```

Pada cara ini, Anda bisa memasukan hasil yang besar secara tepat. Mari kita hitung

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

```
44!/(34!10!)/nilaiC(44,10)
```

```
2481256778
```

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

```
binomial(44,10)//menghitungC(44,10)menggunakanfunksibinomial()
```

```
2481256778
```

Untuk mempelajari lebih lanjut mengenai fungsi tertentu, klik 2 kali di atasnya. Sebagai contoh, mencoba klik 2 kali pada "amp;binomial" dalam baris perintah sebelumnya. Hal ini membuka dokumentasi Maxima sebagaimana yang disediakan oleh penulis program tersebut.

Anda akan mempelajari bahwa hal berikut juga berfungsi.

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

$\text{binomial}(x, 3) / C(x, 3)$

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

Jika Anda ingin mengganti x dengan nilai spesifik apapun, gunakan "with".

$\text{binomial}(x, 3) \text{with } x = 10 / \text{substitus } x = 10 \text{ ke } C(x, 3)$

120

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

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

Seperti yang akan Anda lihat dalam contoh sebelumnya dan berikut, jika Anda memiliki LaTeX yang diinstal, Anda dapat mencetak ekspresi simbolik dengan Latex. Jika tidak, perintah berikut akan mengeluarkan pesan kesalahan.

Untuk mencetak ekspresi simbolik dengan LaTeX, gunakan *didepan amp; (atau Anda dapat menghilangkannya)* jika Anda tidak memiliki LaTeX yang diinstal.

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

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

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

"v := 5; v^2"

25

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

$\text{expand}((1+x)^4), \text{factor}(\text{diff}(\text{$

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

![images/23030630036\_Marcelline

Sekali lagi,

Untuk mempermudah, kami menyimpan solusinya ke variabel simbolik. Variabel simbolik didefinisikan dengan "amp;=".

$\text{fx} = (x+1)/(x^4+1); \text{fx}$

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

Ekspresi simbolik dapat digunakan pada ekspresi simbolik lainnya.

```
factor(diff(fx,x))
```

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

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

```
factor(20!)
2432902008176640000
::: factor(10!)
8 4 2 2 3 5 7
:: factor(20!)
18 8 4 2 2 3 5 7 11 13 17 19
```

Jika Anda seorang ahli di Maxima, Anda mungkin ingin menggunakan sintaks asli Maxima. Anda bisa menggunakan dengan "::::".

```
::: av:gav^2;
2 g
fx = x^3exp(x),fx
3 x x E
```

$$x^3 e^x$$

Variabel semacam itu dapat digunakan dalam ekspresi simbolik lainnya. Perhatikan dalam perintah berikut, sisi kanan amp;= dievaluasi sebelum penugasan ke Fx.

```
(fx with x=5),
5 125 E
```

$$125 e^5$$

```
18551.64488782208
```

```
fx(5)
18551.6448878
```

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

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

```
(fx with x=10)-(fx with x=5), float(
10 5 1000 E - 125 E
2.20079141499189e+7
factor(diff(fx,x,2))
```

$$x (x^2 + 6x + 6) e^x$$

Untuk mendapatkan kode Lateks pada sebuah ekspresi, Anda bisa menggunakan perintah tex.

tex(fx)

$x^3 e^x$

Ekspresi simbolik dapat dievaluasi seperti ekspresi numerik.

fx(0.5)

0.206090158838

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

$fxwithx = 1/2$

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

Penugasan juga bersifat simbolik.

$fxwithx = 1 + t$

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

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

$solve(x^2 + x = 4, x)$

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

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

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

1.56155281281

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

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

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

[-3.23607, 1.23607]

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

Untuk mendapatkan solusi simbolik yang spesifik, Anda dapat menggunakan "with" dan indeks.

$solve(x^2 + x = 1, x), x2 = xwith$

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

!images/23030630036\_Marcelline

Untuk menyelesaikan sistem persamaan, gunakan persamaan vektor. Hasilnya akan berupa solusi vektor.

```
sol = solve([x+y=3,x^2 + y^2 = 5],[x,y]);sol, xywithsol[1]
```

2

!images/23030630036\_Marcelline

Ekspresi simbolik dapat memiliki flag yang menunjukkan perlakuan khusus di Maxima. Beberapa flag dapat digunakan sebagai perintah juga, yang lain tidak. Flag ditambahkan dengan "—" (bentuk yang lebih baik dari "ev(...,flags)")

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

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

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

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

```
factor(
```

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

Fungsi

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

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

```
function f(x) := xsqrt(x^2 + 1)
```

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

```
f(2)
```

```
4.472135955
```

Fungsi ini akan berfungsi pada vektor, mematuhi bahasa matriks Euler, karena ekspresi yang digunakan dalam fungsi tersebut vektorisasi.

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

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

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

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

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

```
0.786151377757
```



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

Anda dapat memanggil fungsi dibawah ini sebagai "...", *jika itu adalah sebuah fungsi dalam inti Euler.*

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

```
sin(45)
```

```
0.707106781187
```

Sebaiknya kita menghapus definisi ulang dari sin.

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

```
0.707106781187
```

Parameter Default

Fungsi numerik dapat memiliki parameter default.

```
function f(x,a=1) := ax2
```

Melewatkan parameter ini menggunakan nilai default.

```
f(4)
```

```
16
```

Mengaturnya menimpa nilai default.

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

```
80
```

Parameter yang ditetapkan juga menimpa parameter default. Hal ini digunakan oleh banyak fungsi Euler seperti plot2d, plot3d.

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

```
16
```

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

```
function f(x) := ax2
```

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

```
24
```

Tetapi parameter yang ditetapkan menimpa nilai global.

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

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

```
20
```

Fungsi simbolik didefinisikan dengan "&#106;=". Mereka didefinisikan dalam Euler dan Maxima, dan bekerja di kedua dunia. Ekspresi definisi dijalankan melalui Maxima sebelum definisi.

```
function g(x) = x3 - xexp(-x);g(x)
```

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

Fungsi simbolik dapat digunakan dalam ekspresi simbolik lainnya.

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

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

!images/23030630036\_Marcelline

Mereka juga dapat digunakan dalam ekspresi numerik. Tentu saja, ini hanya akan berfungsi jika EMT dapat menginterpretasikan semuanya dalam suatu fungsi.

```
g(5+g(1))
178.635099908
```

Mereka dapat digunakan untuk mendefinisikan simbol fungsi atau ekspresi lainnya.

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

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

```
solve(g(x),0.5)
0.703467422498
```

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

```
solve(g,0.5)
0.703467422498
function P(x,n) = (2x-1)^n;P(x,n)
```

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

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

$$(x + 2)^n$$

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

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

```
![images/23030630036_Marcelline
```

```
P(3,4)
625
P(x,4) + Q(x,3),expand(
```

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

```
![images/23030630036_Marcelline
```

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

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

```
![images/23030630036_Marcelline
```

```
![images/23030630036_Marcelline
P(x,4)Q(x,3),expand(
```

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

```
![images/23030630036_Marcelline
```

![images/23030630036\_Marcelline  
 $P(x, 4)/Q(x, 1), \text{expand}(\$

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

![images/23030630036\_Marcelline  
 ![images/23030630036\_Marcelline  
 function f(x) = x<sup>3</sup> - x; f(x)

$$x^3 - x$$

Dengan `:=` fungsi tersebut bersifat simbolik, dan dapat digunakan dalam ekspresi simbolik lainnya.

`integrate(f(x), x)`

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

Dengan `:=` fungsi tersebut bersifat numerik. Contoh yang baik adalah integral tentu seperti

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

yang tidak dapat dievaluasi secara simbolik.

Jika kita mendefinisikan ulang fungsi dengan kata kunci "map", fungsi tersebut dapat digunakan untuk vektor x. Secara internal, fungsi tersebut dipanggil untuk semua nilai x sekali, 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 tersebut dapat dipanggil dengan atau tanpa parameter "base".

`mylog(100), mylog(26.7, 2)`

`2 6.7`

Selain itu, dimungkinkan untuk menggunakan parameter yang ditetapkan.

`mylog(E2, base = E)`

`2`

Seringkali, kita ingin menggunakan fungsi dari vektor di satu tempat, dan untuk elemen individu di tempat lain. Hal ini mungkin dengan parameter vektor.

`function f([a,b]) = a2 + b2 - ab + b; f(a,b), f(x,y)`

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

![images/23030630036\_Marcelline

Fungsi simbolik seperti itu dapat digunakan untuk variabel simbolik.

Tetapi fungsi tersebut juga dapat digunakan untuk vektor numerik.

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

17

Ada juga fungsi-fungsi simbolik murni yang tidak bisa digunakan secara numerik.

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

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

0

![images/23030630036\_Marcelline

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

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

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

Untuk meringkas

\*  $\text{amp};=$  mendefinisikan fungsi simbolik,

\*  $:=$  mendefinisikan fungsi numerik,

\*  $\text{amp};\text{amp};=$  mendefinisikan fungsi simbolik murni.

Menyelesaikan Ekspresi

Ekspresi dapat diselesaikan secara numerik dan simbolik.

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

$solve('x^2 - 2', 1)$

1.41421356237

Ini juga bekerja pada ekspresi simbolik. Ambil fungsi berikut.

$solve(x^2 = 2, x)$

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

$solve(x^2 - 2, x)$

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

$solve(ax^2 + bx + c = 0, x)$

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

$solve([ax + by = c, dx + ey = f], [x, y])$

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

$$px = 4x^8 + x^7 - x^4 - x; px$$

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

Sekarang kita mencari titik dimana nilai polinomialnya 2. Pada solve (), nilai target default y=0 dapat diubah dengan variabel yang ditetapkan.

Kami menggunakan y=2 dan memeriksa dengan evaluasi polinomial pada hasil sebelumnya.

```
solve(px,1,y=2), px(
0.966715594851 2
```

Menyelesaikan ekspresi simbolik dalam bentuk simbolik mengembalikan daftar solusi. Kami menggunakan solver simbolik solve() yang disediakan oleh Maxima.

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

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

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

```
longest sol()
-0.6180339887498949 1.618033988749895
```

Untuk menggunakan solusi simbolik di ekspresi lain, cara termudahnya adalah dengan "with".

```
x^2withsol[1],expand(x^2 - x - 1withsol[2])
```

0

![images/23030630036\_Marcelline

Menyelesaikan sistem persamaan secara simbolik dapat dilakukan dengan persamaan vektor dan solver simbolik solve(). Jawabannya adalah daftar-daftar persamaan.

```
solve([x + y = 2, x^3 + 2y + x = 4], [x, y])
```

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

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

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

dengan a=3.

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

Satu cara untuk meneruskan parameter tambahan ke f() adalah dengan menggunakan daftar dengan nama fungsi dan parameternya (cara lainnya adalah parameter titik koma).

```
solve("f",3,2,y=0.1)
2.54116291558
```

Hal ini juga berfungsi dengan ekspresi. Tetapi kemudian, elemen daftar bernama harus digunakan. (Lebih lanjut tentang daftar dalam tutorial tentang sintaks EMT).

```
solve("x^a - a^x", a = 3, 2, y = 0.1)
2.54116291558
```

Menyelesaikan Pertidaksamaan

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

```
load(fourier_elim)
```

```
C:/Program Files/Euler x64/maxima/share/maxima/5.35.1/share/fourier_elim/fourier_elim.lisp
fourier_elim([x^2 - 1 < 0], [x])/x^2 - 1 < 0
```

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

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

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

```
fourier_elim([x^2 - 10], [x])/x - 1 < 0
```

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

```
fourier_elim([x^6], [x])
```

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

```
fourier_elim([x < 1, x > 1], [x])/tidakmemilikipenyelesaian
```

$$\text{emptyset}$$

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

$$\text{universalset}$$

```
fourier_elim([x^3 - 1 < 0], [x])
```

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

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

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

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

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

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

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

*fourier\_elim*(( $x + y < 5$ )*and*( $x - y \geq 8$ ),  $[x, y]$ )

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

*fourier\_elim*(( $(x + y < 5)$ *and* $x < 1$ )*or*( $x - y \geq 8$ ),  $[x, y]$ )

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

*fourier\_elim*( $[\max(x, y) \leq 6, x \leq 8, \text{abs}(y-1) \leq 12], [x, y]$ )

$[6 \leq x, x \leq 8, y \leq -11]$  *or*  $[8 \leq x, y \leq -11]$  *or*  $[x \leq 8, 13 \leq y]$  *or*  $[x = y, 13 \leq y]$  *or*  $[8 \leq x, x \leq y, 13 \leq y]$  *or*  $[y \leq x, 13 \leq y]$

*fourier\_elim*( $[(x + 6)/(x - 9) \leq 6], [x]$ )

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

### Bahasa Matriks

Dokumentasi inti EMT ini berisi diskusi terperinci tentang bahasa matriks Euler.

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

A=[1,2;3,4]

1 2 3 4

Produk matriks dinotasikan dengan titik.

b=[3;4]

3 4

b' // transpose b

[3, 4]

inv(A) //inverse A

-2 1 1.5 -0.5

A.b //perkalian matriks

11 25

A.inv(A)

1 0 0 1

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

A.A

7 10 15 22

A<sup>2</sup>//perpangkatanelemen2A

1 4 9 16

A.A.A

37 54 81 118

power(A,3) //perpangkatan matriks

37 54 81 118

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

1 1 1 1

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

```

0.333333 0.666667 0.75 1
A
b // hasilkali invers A dan b, A-1b
-2 2.5
inv(A).b
-2 2.5
A
A // A-1A
1 0 0 1
inv(A).A
1 0 0 1
AA // perkalian elemen-elemen matriks seletak
1 4 9 16
Ini bukanlah produk matriks, tetapi perkalian elemen dengan elemen. Hal
yang sama berlaku untuk vektor.
b2 // perkalian elemen – elemen matriks/vektor
9 16
Jika salah satu operand adalah suatu vektor atau skalar, maka akan diper-
luas secara alami.
2A
2 4 6 8
Sebagai contoh, jika operandnya adalah vektor kolom, elemen-elemennya
diterapkan ke semua baris dari A.
[1,2]A
1 4 3 8
Jika itu adalah vektor baris, maka diterapkan ke semua kolom dari A.
A[2,3]
2 6 6 12
Seseorang dapat membayangkan perkalian ini seolah-olah adalah vektor baris
v telah diduplikasi untuk membentuk matriks dengan ukuran yang sama dengan
A.
dup([1,2],2) // dup: menduplikasi/menggandakan vektor [1,2] sebanyak 2
kali (baris)
1 2 1 2
Adup([1,2],2)
1 4 3 8
Hal ini juga berlaku untuk 2 vektor, dimana salah satunya adalah vektor
baris dan yang lainnya adalah vektor kolom. Kami menghitung i*j untuk i,j
dari 1 hingga 5. Triknya adalah mengalikan 1:5 dengan transposenya. Bahasa
matriks Euler secara otomatis menghasilkan tabel nilai.
(1:5)(1:5)' // hasilkali elemen-elemen vektor baris dan vektor kolom
1 2 3 4 5 2 4 6 8 10 3 6 9 12 15 4 8 12 16 20 5 10 15 20 25
Sekali lagi, ingat bahwa ini bukanlah produk matriks!
(1:5).(1:5)' // hasilkali vektor baris dan vektor kolom
55
sum((1:5)(1:5)) // sama hasilnya

```



55

Bahkan operator seperti `lt`; atau `==` bekerja dengan cara yang sama.

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

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

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

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

5

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

Kita mendapatkan vektor 0 dan 1, dimana 1 berarti benar.

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

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

Dari vektor semacam itu, `"nonzeros"` memilih elemen non-nol.

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

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

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

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

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

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

Sebagai contoh, mari kita mencari semua kuadrat dari angka 1 hingga 1000, yang merupakan 5 modulo 11 dan 3 modulo 13.

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

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

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

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

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

112

Fungsi `nonzeros()` hanya berfungsi pada vektor. Untuk matriks, terdapat `mnonzeros()`.

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

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

Hal ini mengembalikan indeks-indeks dari elemen-elemen yang bukan nol dalam matriks.

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

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

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

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

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

Fungsi `mset()` juga dapat mengatur pada indeks menjadi entri dari matriks lain.

```
mset(A,k,-random(size(A)))
0.765761 0.401188 0.406347 -0.126917 -0.122404 -0.691673 0.495975 0.952814
0.548138 -0.483902 0.444255 0.539246
```

Dan ini sangat mungkin untuk mendapatkan elemen dalam vektor.

```
mget(A,k)
[0.267829, 0.13673, 0.390567, 0.006085]
```

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

```
ex=extrema(A)
0.267829 4 0.765761 1 0.13673 1 0.952814 4 0.006085 2 0.548138 1
```

Kita dapat menggunakannya untuk mengekstrak nilai maksimal di setiap baris.

```
ex[,3]
[0.765761, 0.952814, 0.548138]
```

Ini, tentu saja, ini sama seperti fungsi `max()`.

```
max(A)
[0.765761, 0.952814, 0.548138]
```

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

```
j=(1:rows(A))'-ex[,4], mget(-A,j)
1 1 2 4 3 1 [-0.765761, -0.952814, -0.548138]
```

Fungsi Matriks Lainnya (Membangun Matriks)

Untuk membentuk matriks, kita dapat menumpuk 1 matriks diatas yang lain. Jika keduanya tidak memiliki jumlah kolom yang sama, yang lebih pendek akan terisi dengan 0.

```
v=1:3; v_v
1 2 3 1 2 3
```

Demikian pula, kita dapat menempelkan matriks ke sisi yang lain secara berdampingan, jika keduanya memiliki jumlah baris yang sama.

```
A=random(3,4); A—v'
0.032444 0.0534171 0.595713 0.564454 1 0.83916 0.175552 0.396988 0.83514
2 0.0257573 0.658585 0.629832 0.770895 3
```

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

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

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

Ini memungkinkan untuk membuat matriks dari vektor baris dan kolom.

```
[v;v]
```

```

1 2 3 1 2 3
[v',v']
1 1 2 2 3 3
Tujuan utama dari ini adalah untuk menginterpretasikan vektor ekspresi
untuk vektor kolom.
"[x,x^2]"(v')
1 1 2 4 3 9
Untuk mendapatkan ukuran A, kita dapat mengikuti fungsi berikut.
C=zeros(2,4); rows(C), cols(C), size(C), length(C)
2 4 [2, 4] 4
Untuk vektor, terdapat length().
length(2:10)
9
Terdapat banyak lagi fungsi lain, untuk menghasilkan matriks.
ones(2,2)
1 1 1 1
Hal ini juga dapat digunakan dengan satu parameter. Untuk mendapatkan
vektor dengan angka lain selain 1, gunakan yang berikut.
ones(5)6
[6, 6, 6, 6, 6]
Juga matriks angka acak dapat dihasilkan dengan random (distribusi ser-
agam) atau normal (distribusi Gauß).
random(2,2)
0.66566 0.831835 0.977 0.544258
Berikut adalah fungsi lain yang berguna, yang menata ulang elemen matriks
menjadi matriks lain.
redim(1:9,3,3) // menyusun elemen2 1, 2, 3, ..., 9 ke bentuk matriks 3x3
1 2 3 4 5 6 7 8 9
Dengan fungsi berikut, kita dapat menggunakan fungsi ini dan fungsi dup
untuk menulis fungsi rep(), yang mengulangi vektor n kali.
function rep(v,n) := redim(dup(v,n),1,ncols(v))
Mari kita coba.
rep(1:3,5)
[1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3]
Fungsi multdup() menduplikasi elemen dari sebuah vektor.
multdup(1:3,5), multdup(1:3,[2,3,2])
[1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3] [1, 1, 2, 2, 2, 3, 3]
Fungsi flipx() dan flipy() membalik urutan baris atau kolom matriks. Artinya,
fungsi flipx() membalik secara horizontal.
flipx(1:5) //membalik elemen2 vektor baris
[5, 4, 3, 2, 1]
Untuk rotasi, Euler mempunyai rotright() dan rotright().
rotright(1:5) // memutar elemen2 vektor baris
[2, 3, 4, 5, 1]
Fungsi khusus drip (v,i) menghapus elemen-elemen dengan indeks dalam i
dari vektor v.

```

```

drop(10:20,3)
[10, 11, 13, 14, 15, 16, 17, 18, 19, 20]

```

Perhatikan bahwa vektor  $i$  dalam `drop(v,i)` mengacu pada indeks elemen dalam  $v$ , bukan nilai dari elemen tersebut. Jika Anda ingin menghapus elemen, pertama-tama Anda butuh untuk mencari elemen. Fungsi `indexof(v,x)` dapat digunakan untuk mencari elemen  $x$  dalam vektor  $v$  yang terurut.

```

v=primes(50), i=indexof(v,10:20), drop(v,i)
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47] [0, 5, 0, 6, 0, 0, 0, 7, 0,
8, 0] [2, 3, 5, 7, 23, 29, 31, 37, 41, 43, 47]

```

Seperti yang anda lihat, tidak ada salahnya memasukan indeks diluar jangkauan (seperti 0), indeks ganda, atau indeks yang tidak terurut.

```

drop(1:10,shuffle([0,0,5,5,7,12,12]))
[1, 2, 3, 4, 6, 8, 9, 10]

```

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

Kita mulai dengan matriks identitas.

```

A=id(5) // matriks identitas 5x5
1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1

```

Kemudian kita atur diagonal terendah (-1) ke 1:4.

```

setdiag(A,-1,1:4) //mengganti diagonal di bawah diagonal utama
1 0 0 0 0 1 1 0 0 0 0 2 1 0 0 0 0 3 1 0 0 0 0 4 1

```

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

Berikut adalah fungsi yang mengembalikan matriks tri-diagonal.

```

function tridiag (n,a,b,c) := setdiag(setdiag(bid(n),1,c),-1,a); ... tridiag(5,1,2,3)
2 3 0 0 0 1 2 3 0 0 0 1 2 3 0 0 0 1 2 3 0 0 0 1 2

```

Diagonal dari matriks juga dapat diekstrak dari matriks. Untuk mendemonstrasikan hal ini, kita menata ulang vektor 1:9 menjadi matriks 3x3.

```

A=redim(1:9,3,3)
1 2 3 4 5 6 7 8 9

```

Sekarang kita bisa mengekstrak diagonal.

```

d=getdiag(A,0)
[1, 5, 9]

```

Sebagai contoh kita dapat membagi matriks dengan diagonalnya. Bahasa matriks menangani agar vektor kolom  $d$  ditetapkan pada matriks baris demi baris.

```

fraction A/d'
1 2 3 4/5 1 6/5 7/9 8/9 1

```

Vektorisasi

Hampir semua fungsi di Euler bekerja untuk matriks dan vektor, kapanpun al ini tidak masuk akal.

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

```

sqrt(1:3)
[1, 1.41421, 1.73205]

```

Jadi anda bisa dengan mudah membuat tabel nilai. Ini adalah salah satu cara untuk mem-plot fungsi (alternatifnya menggunakan Ekspresi).

```
x=1:0.01:5; y=log(x)/x^2; //terlalupanjanguntukditampilkan
```

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

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

$$s = t^3 - t$$

```
t=-1:0.1:1; s=t^3 - t
[0, 0.171, 0.288, 0.357, 0.384, 0.375, 0.336, 0.273, 0.192, 0.099, 0, -0.099,
-0.192, -0.273, -0.336, -0.375, -0.384, -0.357, -0.288, -0.171, 0]
```

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

Sebagai contoh, vektor kolom dikalikan dengan vektor baris akan menghasilkan matrks, jika operator ditetapkan. Dalam hal berikut, v' adalah vektor yang ditranspos (vektor kolom).

```
shortest (1:5)(1:5)'
1 2 3 4 5 2 4 6 8 10 3 6 9 12 15 4 8 12 16 20 5 10 15 20 25
```

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

```
(1:5).(1:5)'
55
```

Secara default, vektor baris dicetak dalam format ringkas.

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

Untuk matriks, operator khusus "." menandakan perkalian matriks, dan "A'" menandakan transposisi. Matriks 1x1 dapat digunakan seperti angka real.

```
v:=[1,2]; v.v',
5 25
```

Untuk mentranspose matriks, kita dapat menggunakan tanda apostrof.

```
v=1:4; v'
1 2 3 4
```

Sehingga kita dapat menghitung matriks A dikali vektor b.

```
A=[1,2,3,4;5,6,7,8]; A.v'
30 70
```

Perhatikan bahwa v tetap vektor baris. Jadi v'.v berbeda dengan v.v'.

```
v'.v
1 2 3 4 2 4 6 8 3 6 9 12 4 8 12 16
```

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

```
v.v'
30
```

Terdapat juga fungsi norm (bersama dengan banyaknya fungsi aljabar linear lainnya).

$\text{norm}(v)^2$

30

Operator dan fungsi mematuhi bahasa matriks Euler.

Berikut adalah ringkasan aturannya.

\* Suatu fungsi yang diterapkan pada vektor atau matriks diterapkan \* pada setiap elemennya.

\* Sebuah operator yang beroperasi pada 2 matriks berukuran sama \* diterapkan secara berpasangan pada elemen-elemen matriks tersebut.

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

Sebagai contoh, nilai skalar kali vektor mengalikan nilai dengan setiap elemen vektor. Atau matriks kali vektor (dengan \*, bukan .) memperluas vektor ke ukuran matriks dengan menduplikatnya.

Berikut adalah kasus sederhana dengan operator ·

$[1,2,3]^2$

[1, 4, 9]

Berikut adalah kasus yang lebih rumit. Sebuah vektor baris kali sebuah vektor kolom memperluas keduanya dengan duplikasi.

$v:=[1,2,3]; vv'$

1 2 3 2 4 6 3 6 9

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

$v.v'$

14

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

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

$vv, \text{sum}(vv), \text{cumsum}(vv)$

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

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

1:4, 1:2:10

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

Untuk menggabungkan matriks dan vektor, terdapat operator "—" dan "·".

$[1,2,3] - [4,5], [1,2,3] \cdot 1$

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

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

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

6

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

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

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

Indeks juga dapat berupa vektor baris indeks. : menunjukan semua indeks.

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

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

Bentuk singkat untuk : adalah dengan menghilangkan indeks sepenuhnya.

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

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

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

```
A4
```

```
4
```

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

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

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

Untuk menggunakan matriks untuk tabel, mari kita reset ke format default, dan menghitung tabel nilai sinus dan cosinus. 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
```

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

Dalam contoh berikut, kami menghitung  $t[j]^i$  untuk  $i$  dari 1 hingga  $n$ . Kami mendapatkan matriks, di mana setiap

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

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

Integrasi numerik `integrate()` hanya berfungsi untuk batas interval skalar. Jadi kita perlu memvektorisasinya

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

Kata kunci "map" memvektorisasi fungsi. Fungsi sekarang akan bekerja untuk vektor angka.

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

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

Sub-Matriks and Elemen Matriks

Untuk mengakses elemen matriks, gunakan notasi kurung siku.

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

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

Kita dapat mengakses seluruh baris matriks.

```
A[2]
```

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

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

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

```
2
```

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

```
A[2,]
```

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

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

Disini kita menginginkan baris pertama dan kedua dari A.

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

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

Kita bahkan dapat mengubah urutan A menggunakan vektor indeks. Secara tepat, kita tidak mengubah A disini, tetapi menghitung versi A yang diurutkan ulang.

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

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

Trik indeks juga berfungsi dengan kolom.

Contoh ini memilih semua baris dari A serta kolom kedua dan ketiga.

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

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

Sebagai singkatan, ":" melambangkan semua indeks baris atau kolom.

```
A[:,3]
```

```
3 6 9
```

Sebagai alternatif, biarkan indeks pertama kosong.

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

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

Kita juga bisa mendapatkan baris terakhir pada A.

```
A[-1]
```

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

Sekarang mari kita ubah elemen-elemen dari A dengan menetapkan submatriks dari A ke beberapa nilai. Ini sebenarnya akan mengubah matriks A yang tersimpan.

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

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

Kita juga dapat menetapkan nilai ke baris A.

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

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

Kita bahkan bisa menetapkan nilai ke submatriks jika ukurannya sesuai.

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

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

Selain itu, beberapa pintasan diizinkan.



```
A[1:2,1:2]=0
0 0 -1 0 0 6 7 8 9
```

Perhatian: Indeks yang berada di luar batas akan mengembalikan matriks kosong, atau pesan kesalahan, tergantung pada pengaturan sistem. Secara default, akan muncul pesan kesalahan. Namun, ingatlah bahwa indeks negatif dapat digunakan untuk mengakses elemen matriks dengan menghitung dari akhir.

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

Mengurutkan dan Mengocok

Fungsi `sort()` mengurutkan vektor baris

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

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

Seringkali perlu untuk mengetahui indeks dari vektor yang diurutkan dalam vektor aslinya. Ini bisa digunakan untuk mengurutkan vektor lain dengan cara yang sama.

Mari kita acak vektor.

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

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

Indeks ditampung berisi urutan yang tepat dari v.

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

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

Ini juga berlaku vektor string.

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

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

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

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

Dapat anda lihat, posisi entri ganda cukup acak.

```
ind
```

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

Fungsi `unique` mengembalikan daftar terurut dari elemen-elemen unik dalam sebuah vektor.

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

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

Ini juga bekerja pada vektor string.

```
unique(s)
```

```
a aa d e
```

Aljabar Linear

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

Untuk sistem linear  $Ax=b$ , anda dapat menggunakan algoritma Gauss, matriks invers atau kecocokan linear. Operasi `\` menggunakan versi dari algoritma Gauss.

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

```
b
```

```
-4 4.5
```

Untuk contoh lain, kita menghasilkan matriks 200x200 dan jumlah baris-barisnya. Kemudian kita menyelesaikan  $Ax=b$  menggunakan matriks invers. Kita mengukur kesalahan sebagai deviasi maksimum dari semua elemen dari 1, yang tentu saja adalah solusi yang benar.

```
A=normal(200,200); b=sum(A); longest totalmax(abs(inv(A).b-1))
1.177724584522366e-12
```

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

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

Determinan matriks ini adalah 0.

```
det(A)
0
```

Matriks simbolik

Maxima memiliki matriks simbolik. Tentu saja, maxima dapat digunakan untuk masalah aljabar linear sederhana. Kita dapat mendefinisikan matriks untuk Euler dan Maxima dengan `:=`, dan kemudian menggunakannya dalam ekspresi simbolik. Bentuk biasa [...] untuk mendefinisikan matriks dapat digunakan di Euler untuk mendefinisikan matriks simbolik.

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

$$(a) 111a111a$$

```
det(A),factor(
```

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

```
![images/23030630036_Marcelline
invert(A)witha = 0
```

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

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

$$(1) ab2$$

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

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

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

```
![images/23030630036_Marcelline
```

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

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

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

Untuk mengekstrak vektor eigen tertentu, diperlukan pemilihan indeks yang hati-hati.

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

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

[1, - 1]

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

A(a=4,b=5)

1 4 5 2

Pada ekspresi simbolik, gunakan with.

*Awith*[a = 4, b = 5]

$$(1) 452$$

Akses ke baris matriks simbolik bekerja sama seperti pada matriks numerik.

A[1]

$$[1, a]$$

Sebuah ekspresi simbolik dapat berisi penugasan. Dan hal itu mengubah matriks A.

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

$$(t) + 1ab2$$

Terdapat fungsi simbolik di Maxima yang bertujuan untuk membentuk vektor dan matriks. Untuk ini, lihat dokumentasi Maxima atau tutorial tentang Maxima di EMT.

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

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

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

$$(-) 21 \frac{3}{2} - \frac{1}{2}$$

![images/23030630036\_Marcelline

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

*invert*(B)()

-2 1 1.5 -0.5

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

Perhatikan bahwa dengan `amp;:=` , matriks B didefinisikan secara simbolik pada ekspresi simbolik dan numerik pada ekspresi numerik. Jadi kita dapat menggunakannya disini.

```
longest B.xinv(B)
1 0 0 1
```

Sebagai contoh pada nilai eigen A dapat dihitung secara numerik.

```
A=[1,2,3;4,5,6;7,8,9]; real(eigenvalues(A))
[16.1168, -1.11684, 0]
```

Atau simbolik. Lihat tutorial tentang Maxima untuk detail hal ini.

```
eigenvalues(@A)
```

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

Nilai Numerik dalam Ekspresi Simbolik

Ekspresi simbolik hanyalah sebuah string yang berisikan sebuah ekspresi. Jika kita ingin mendefinisikan nilai baik ekspresi simbolik dan ekspresi numerik, kita harus menggunakan `"amp;:="`.

```
A := [1,pi;4,5]
1 3.14159 4 5
```

Masih ada perbedaan antara bentuk numerik dan simbolik. Ketika men-transfer matriks ke bentuk simbolik, pendekatan pecahan untuk bilangan real akan digunakan.

```
A
```

$$(1) \frac{1146408}{364913} 45$$

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

```
mxmset(A); A
```

$$(1) 3.14159265358979345$$

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

```
bfloat(sqrt(2)),float(sqrt(2))
```

$$1.414213562373095$$

!images/23030630036\_Marcelline

Presisi angka floating poin besar dapat diubah.

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

```
3.14159265358979323846264338327950288419716939937510582097494 45923078164062862089986280348253
```

Variabel numerik dapat digunakan pada ekspresi simbolik manapun menggunakan `"@var"`.

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

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

−5.424777960769379

#### Demo - Tingkat Bunga

Dibawah ini, kita menggunakan Euler Math Toolbox (EMT) untuk menghitung tingkat bunga. Kami melakukannya secara numerik dan simbolik untuk menunjukkan pada Anda bagaimana Euler dapat bekerja dalam menangani masalah kehidupan nyata.

Misalkan Anda memiliki modal awal sebesar 5000(katakan dalam dolar).

```
K=5000
```

```
5000
```

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

```
K*1.03
```

```
5150
```

Euler juga akan memahami sintaks berikut.

```
K+K*3
```

```
5150
```

Namun, lebih mudah menggunakan faktor.

```
q=1+3
```

```
1.03 5150
```

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

```
K*q^10
```

```
6719.58189672
```

Untuk keperluan kita, kita bisa mengatur format ke 2 digit setelah titik desimal.

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

```
6719.58
```

Mari kita cetak hasil tersebut untuk dibulatkan menjadi 2 digit dalam sebuah kalimat lengkap.

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

```
Starting from 5000you get6719.58.
```

Bagaimana jika kita ingin mengetahui hasil antara tahun pertama hingga tahun 9? Untuk ini, bahasa matriks Euler sangat membantu. Anda tidak perlu untuk menuliskan loop, tetapi cukup memasukan

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

```
Real 1 x 11 matrix
```

```
5000.00 5150.00 5304.50 5463.64 ...
```

Bagaimana cara kerja keajaiban ini? Pertama, ekspresi 0:10 mengembalikan vektor bilangan bulat.

```
short 0:10
```

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

Kemudian semua operator dan fungsi di Euler dapat diterapkan ke elemen-elemen vektor satu per satu. Jadi,

short q(0 : 10)  
 [1, 1.03, 1.0609, 1.0927, 1.1255, 1.1593, 1.1941, 1.2299, 1.2668, 1.3048, 1.3439]  
 adalah sebuah vektor dari faktor  $q^0$  ke  $q^{10}$ . Inidikalikandengan K, dankitamendapatkanvektornilai–  
 nilai.

VK=Kq(0 : 10);

Tentu saja, cara realistik untuk menghitung tingkat bunga ini adalah dengan membulatkan ke sen terdekat setelah tiap tahun. Mari kita menambahkan fungsi untuk ini.

function oneyear (K) := round(Kq,2)

Mari kita membandingkan hasil keduanya, dengan dan tanpa pembulatan.

longest oneyear(1234.57), longest 1234.57q

1271.61 1271.6071

Sekarang tidak ada formula sederhana untuk tahun n, dan kita harus melakukan loop selama tahun-tahun tersebut. Euler menyediakan banyak solusi untuk ini.

Cara termudah yaitu dengan fungsi iterate, yang mengulangi fungsi tertentu sejumlah kali.

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

Real 1 x 11 matrix

5000.00 5150.00 5304.50 5463.64 ...

Kita dapat mencetaknya dengan cara yang ramah, menggunakan format kita dengan tempat desimal tetap.

VKr'

5000.00 5150.00 5304.50 5463.64 5627.55 5796.38 5970.27 6149.38 6333.86  
 6523.88 6719.60

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

VKr[2], VKr[1:3]

5150.00 5000.00 5150.00 5304.50

Secara mengejutkan, kita juga dapat menggunakan vektor indeks. Ingat bahwa 1:3 membentuk vektor [1,2,3].

Mari kita bandingkan elemen terakhir dari nilai yang dibulatkan dengan nilai lengkapnya.

VKr[-1], VK[-1]

6719.60 6719.58

Perbedaannya sangat kecil.

Menyelesaikan Persamaan.

Sekarang kita menggunakan fungsi yang lebih maju, yang menambahkan sejumlah uang tertentu setiap tahun.

function onepay (K) := Kq+R

Kita tidak perlu menentukan q atau R untuk mendefinisikan suatu fungsi. Hanya saat menjalankan perintah, kita harus mendefinisikan nilai-nilai ini. Kami pilih R=200.

R=200; iterate("onepay",5000,10)

Real 1 x 11 matrix

5000.00 5350.00 5710.50 6081.82 ...

Bagaimana jika kita mengurangi jumlah yang sama setiap tahun?

```
R=-200; iterate("onipay",5000,10)
```

```
Real 1 x 11 matrix
```

```
5000.00 4950.00 4898.50 4845.45 ...
```

Kita lihat bahwa uang berkurang. Jelas, karena kita hanya mendapatkan 150 dari bunga pada tahun pertama, tetapi mengurangi 200, kita kehilangan uang setiap tahun.

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

```
VKR=iterate("onipay",5000,50)
```

```
Real 1 x 51 matrix
```

```
5000.00 4950.00 4898.50 4845.45 ...
```

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

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

```
48.00
```

Alasannya adalah bahwa `nonzeros(VKR<0)` mengembalikan vektor indeks `i`, dimana `VKR[i]<0`, dan `min` menghitung indeks minimal.

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

Fungsi `iterate()` memiliki satu trik lagi. Fungsi ini dapat menerima kondisi akhir sebagai argumen. Kemudian mengembalikan nilai dan jumlah iterasi.

```
x,n=iterate("onipay",5000,till="x<0"); x, n,
```

```
-19.83 47.00
```

Mari kita coba untuk menjawab pertanyaan yang lebih ambigu. Misalkan kita tahu bahwa nilai uang adalah 0 setelah 50 tahun. Berapa tingkat bunga yang diperlukan?

Pada pertanyaan ini, kita hanya dapat menjawabnya dengan numerik. Dibawah, kita akan menurunkan rumus yang diperlukan. Kemudian Anda akan melihat bahwa tidak ada rumus mudah untuk tingkat bunga. Tetapi untuk sekarang, kita bertujuan untuk solusi numerik.

Langkah pertama adalah untuk mendefinisikan fungsi yang melakukan iterasi sebanyak `n` kali. Kita tambahkan semua paramater ke fungsi ini.

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

Iterasi adalah seperti dibawah ini

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

Tetapi kita tidak lagi menggunakan nilai global `R` pada ekspresi ini. Fungsi seperti `iterate()` memiliki trik khusus di Euler. Anda dapat melewati nilai variabel pada ekspresi sebagai parameter titik koma. Pada kasus ini `P` dan `R`.

Selain itu, kita hanya tertarik pada nilai akhir. Sehingga kita mengambil indeks `[-1]`.

Mari kita lakukan uji coba.

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

```
-19.83
```

Sekarang kita dapat menyelesaikan permasalahan kita

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

3.15

Rutin solve menyelesaikan expression=0 untuk variabel x. Jawabannya adalah 3.15 Fungsi solve() selalu dibutuhkan untuk nilai awal.

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

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

-336.08

Perhatikan bahwa Anda tidak dapat menyelesaikan untuk jumlah tahun, karena fungsi kita mengasumsikan n sebagai nilai bilangan bulat.

Solusi Simbolik untuk Masalah Tingkat Bunga

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

```
function op(K) = Kq+R; op(K)
```

$$R + q K$$

Kita bisa melakukan iterasi pada fungsi ini.

```
op(op(op(op(K))))),expand(
```

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

!images/23030630036\_Marcelline

Kita melihat sebuah pola. Setelah n periode, kita memiliki

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

Rumus ini adalah rumus untuk jumlah geometri, yang dikenal oleh Maxima.

```
sum(q^k, k, 0, n - 1);
```

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

Ini agak rumit. Jumlahnya dievaluasi dengan flag "simpsum" untuk menyederhanakannya menjadi bentuk pecahan.

Mari kita buat fungsi untuk ini.

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

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

Fungsi ini melakukan hal yang sama seperti fungsi f kita sebelumnya. Namun ini lebih efektif.

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



-19.82504734650985 -19.82504734652684

Kita sekarang dapat menggunakannya untuk menanyakan waktu  $n$ . Kapan modal kita habis? Tebakan awal kita adalah 30 tahun.

`solve("fs(5000,-330,3,x)",30)`  
20.51

Jawaban ini mengatahakan bahwa ini akan menjadi negatif setelah 21 tahun.

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

Misalkan kita mendapatkan pinjaman sebesar  $K$ , dan membayar  $n$  pembayaran sebesar  $R$  (memulai setelah 1 tahun) meninggalkan utang sisa  $Kn$  (pada saat pembayaran terakhir). Rumus untuk ini jelas adalah

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

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

Biasanya rumus ini diberikan dalam bentuk

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

`equ = (equ with P=100i); equ`

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

Kita dapat menyelesaikan untuk besaran simbolik  $R$ .

`solve(equ, R)`

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

Seperti yang anda lihat pada rumus, fungsi ini menghasilkan error floating point untuk  $i=0$ . Meskipun demikian, Euler masih memplotnya.

Tentu saja, kita memiliki batas berikut.

`limit(R(5000,0,x,10),x,0)`

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

Jelas, tanpa bunga kita harus membayar 10 cicilan sebesar 500.

Persamaan ini juga dapat diselesaikan untuk  $n$ . Hasilnya terlihat lebih baik jika kita menerapkan beberapa penyederhanaan pada persamaan tersebut.

`fn = solve(equ,n) — ratsimp; fn`

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

LATIHAN SOAL

R.2

1) Hitunglah nilai dari

$$2^6 \cdot 2^{-3} \div 2^{10} \div 2^{-8}$$

$$\text{expand}(2^6 2^{-3} / 2^{10} / 2^{-8})$$

2

Hasil yang didapat dari perhitungan tersebut yaitu 2, sama seperti apabila dihitung secara manual

2) Hitunglah nilai dari

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

$$\text{expand}(4((8-6)^2) - (43) + (28)) / (3^1 + 19^0)$$

5

Hasil yang didapat dari perhitungan tersebut yaitu 5, sama seperti apabila dihitung secara manual

3) Sederhanakanlah

$$\left[ \frac{(3x^a y^b)^3}{(-3x^a y^b)^2} \right]^2$$

$$\text{expand}((3x^a y^b)^3 / (-3x^a y^b)^2)^2$$

$$9x^{2a}y^{2b}$$

Hasil yang didapat dari penyederhanaan tersebut yaitu  $9x^{(2a)}y^{(2b)}$ , sama seperti apabila dihitung secara manual

4) Hitunglah nilai dari

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

$$\text{expand}((4(8-6)^2 + 4)(3-28)) / 2^2(2^3 + 5)$$

-845

Hasil yang didapat dari perhitungan tersebut yaitu -845, sama seperti apabila dihitung secara manual.

5) Sederhanakanlah

$$\left[ \left( \frac{x^r}{y^t} \right)^2 \left( \frac{x^{2r}}{y^{4t}} \right)^{-2} \right]^{-3}$$

$$\text{expand}([(x^r / y^t)^2 (x^{(2r)} / y^{(4t)})^{-2}]^{-3})$$

$$\left[ \frac{x^{6r}}{y^{18t}} \right]$$

Hasil yang didapat dari penyederhanaan tersebut yaitu  $x^{(6r)}/y^{(18t)}$ , *sama seperti apabila di hitung secara manual*

R.3

1) Lakukan operasi berikut

$$\begin{aligned} & (2x + 3y + z - 7) + (4x - 2y - z + 8) + (-3x + y - 2z - 4) \\ & \text{expand}(2x + 3y + z - 7) + (4x - 2y - z + 8) + (-3x + y - 2z - 4) \\ & -2z + 2y + 3x - 3 \end{aligned}$$

Dari operasi tersebut akan didapat hasil  $-2z+2y+3x-3$

2) Sederhanakan operasi berikut

$$\begin{aligned} & (3a^2)(-7a^4) \\ & \text{expand}((3a^2)(-7a^4)) \\ & -7203a^2 \end{aligned}$$

Dari operasi tersebut akan didapat hasil  $-7203a^2$

3) Hitung penyederhanaan dari

$$\begin{aligned} & (2x + 3y + 4)(2x + 3y - 4) \\ & \text{expand}((2x + 3y + 4)(2x + 3y - 4)) \\ & 9y^2 + 12xy + 4x^2 - 16 \end{aligned}$$

Dari operasi tersebut akan didapat hasil  $9y^2 + 12xy + 4x^2 - 16$

4) Hitung penyelesaian dari

$$\begin{aligned} & (m + 1)(m - 1) \\ & \text{expand}((m + 1)(m - 1)) \\ & m^2 - 1 \end{aligned}$$

Dari operasi tersebut akan didapat hasil  $m^2 - 1$

5) Hitung penyelesaian dari

$$\begin{aligned} & (x + 1)(x - 1)(x^2 + 1) \\ & \text{expand}((x + 1)(x - 1)(x^2 + 1)) \\ & x^4 - 1 \end{aligned}$$

Dari operasi tersebut akan didapat hasil  $x^4 - 1$

R.4

1) Hitunglah faktor dari

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

$$\text{solve}(t^2 + 8t + 15)$$

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

Dari operasi tersebut akan didapat faktor berupa t=-3 atau t=-5

2) Tentukan faktor dari

$$z^2 - 81$$

$$\text{solve}(z^2 - 81)$$

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

Dari operasi tersebut akan didapat faktor berupa z=-9 atau z=9

3) Tentukan faktor kuadrat dari

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

$$\text{solve}(25ab^4 - 25az^4)$$

Maxima said: solve: more unknowns than equations. Unknowns given :  
[z,b,a] Equations given:  $[25*a*b^4 - 25*a*z^4]$  — *anerror.Todebugthis try :*  
*debugmode(true);*

Error in: *amp; solve(25\*a\*b^4 - 25\*a\*z^4)...*

Dari operasi tersebut tidak akan didapat faktor karena perbedaan dalam sukunya.

4) Tentukan faktor dari

$$4t^3 + 108$$

$$\text{solve}(4t^3 + 108)$$

$$\left[ t = \frac{3 - 3^{\frac{3}{2}}i}{2}, t = \frac{3^{\frac{3}{2}}i + 3}{2}, t = -3 \right]$$

Dari operasi tersebut akan didapat 3 faktor karena merupakan perpangkatan 3

5) Tentukan faktor dari

$$x^2 - 5x + \frac{25}{4}$$

$$\text{solve}(x^2 + 5x + 25/4)$$

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

Dari operasi tersebut hanya akan mendapat 1 faktor.

R.5

1) Selesaikan persamaan berikut

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

$$\text{solve}(7(3x + 6) = 11 - (x + 2))$$

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

Dari operasi tersebut akan didapat penyederhanaan berupa  $x = -3/2$

2) Selesaikan persamaan berikut

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

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

$$[x = 10]$$

Dari operasi tersebut akan didapat penyederhanaan berupa  $x = 10$

3) Selesaikan

$$x^2 - 36 = 0$$

$$\text{solve}(x^2 - 36 = 0)$$

$$[x = -6, x = 6]$$

Dari operasi tersebut akan didapat penyederhanaan berupa  $x = -6$  dan  $x = 6$  karena 36 merupakan kuadrat dari -6 dan 6

4) Selesaikan

$$st = t - 4$$

$$\text{solve}(st = t - 4, t)$$

$$[t = st + 4]$$

Maka akan didapat  $t =$  karena yang dicari adalah nilai  $t$

5) Selesaikan

$$3[5 - 3(4 - t)] - 2 = 5[3(5t - 4) + 8] - 26$$

$$\text{solve}(3(5 - 3(4 - t)) - 2 = 5(3(5t - 4) + 8) - 26)$$

$$\left[ t = \frac{23}{66} \right]$$

Dari operasi tersebut akan didapat penyederhanaan berupa  $t = 23/66$

R.6

1) Sederhanakan

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

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

$$[x = -2]$$

Dari operasi tersebut akan didapat penyederhanaan berupa  $x = -2$

2) Sederhanakan

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

*solve*(( $6 - x$ )/ $x^2 - 36$ )

$$\left[ x = \frac{-\sqrt{865} - 1}{72}, x = \frac{\sqrt{865} - 1}{72} \right]$$

Dari operasi tersebut akan didapat penyederhanaan berupa akar

3) Kalikan atau bagi, dan jika mungkin, sederhanakan

$$\frac{r - s}{r + s} \cdot \frac{r^2 - s^2}{(r - s)^2}$$

*expand*(( $(r - s)/(r + s)$ )( $(r^2 - s^2)/(r - s)^2$ ))

$$\frac{r^2}{r^2 - s^2} - \frac{s^2}{r^2 - s^2}$$

*solve*(( $r^2/(r^2 - s^2)$ ) - ( $s^2/(r^2 - s^2)$ ))

Maxima said: solve: more unknowns than equations. Unknowns given :  
[s,r] Equations given: [ $r^2/(r^2 - s^2) - s^2/(r^2 - s^2)$ ] - *anerror.Todebugthis*try :  
*debugmode(true)*;

Error in: *amp; solve*(( $r^2/(r^2 - s^2)$ ) - ( $s^2/(r^2 - s^2)$ ))...

Dari operasi tersebut tidak didapat penyederhanaannya

4) Tambah atau kurang, dan jika mungkin sederhanakan

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

*expand*(( $7/5x$ ) + ( $3/5x$ ))

$$2x$$

Dari operasi tersebut akan didapat penyederhanaan berupa  $2x$

5) Sederhanakan

$$\frac{y^5 - 5y^4 + 4y^3}{y^3 - 6y^2 + 8y}$$

*expand*(( $y^5 - 5y^4 + 4y^3$ )/( $y^3 - 6y^2 + 8y$ ))

$$\frac{y^5}{y^3 - 6y^2 + 8y} - \frac{5y^4}{y^3 - 6y^2 + 8y} + \frac{4y^3}{y^3 - 6y^2 + 8y}$$

$$\text{solve}((y^5 - 5y^4 + 4y^3)/(y^3 - 6y^2 + 8y))$$

$$[y = 0, y = 1]$$

R.7

1) Kalikan dan sederhanakan:

$$\frac{x^2 + x - 6}{x^2 + 8x + 15} \cdot \frac{x^2 - 25}{x^2 - 4x + 4}$$

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

$$\frac{x^4}{x^4 + 4x^3 - 13x^2 - 28x + 60} + \frac{x^3}{x^4 + 4x^3 - 13x^2 - 28x + 60} - \frac{31x^2}{x^4 + 4x^3 - 13x^2 - 28x + 60} - \frac{25x}{x^4 + 4x^3 - 13x^2 - 28x + 60}$$

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

$$[x = 5]$$

2) Kalikan

$$(x^n + 10)(x^n - 4)$$

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

$$x^{2n} + 6x^n - 40$$

3) Kalikan

$$(a^n - b^n)^3$$

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

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

4) Kalikan

$$(y^b - z^c)(y^b + z^c)$$

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

$$y^{2b} - z^{2c}$$

5) Kalikan

$$(t^a + t^{-a})^2$$

$$\text{expand}((t^a + t^{-a})^2)$$

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

2.3

1) Diberikan

$$f(x) = 3x + 1$$

$$g(x) = x^2 - 2x - 6$$

$$h(x) = x^3$$

a. Tentukan nilai

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

function f(x):= 3x+1  
 function g(x):= x<sup>2</sup> - 2x - 6  
 function h(x):= x<sup>3</sup>

3.1

1) Sederhanakan

$$(-5 + 3i) + (7 + 8i)$$

$$\text{expand}((-5 + 3i) + (7 + 8i))$$

$$11i + 2$$

2) Sederhanakan

$$(10 + 7i) - (5 + 3i)$$

$$\text{expand}((10 + 7i) - (5 + 3i))$$

$$4i + 5$$

3) Sederhanakan

$$\sqrt{-4} \cdot \sqrt{-36}$$

$$\text{expand}(\text{sqrt}(-4)\text{sqrt}(-36))$$

$$-12$$

4) Sederhanakan

$$(3 + \sqrt{-16}) + (2 + \sqrt{-25})$$

$$\text{expand}((3 + \text{sqrt}(-16)) + (2 + \text{sqrt}(-25)))$$

$$9i + 5$$



5) Sederhanakan

$$\frac{3+2i}{1-i} + \frac{6+2i}{1-i}$$
$$\text{expand}(((3+2i)/(1-i)) + ((6+2i)/(1-i)))$$
$$\frac{4i}{1-i} + \frac{9}{1-i}$$

3.2

1) Selesaikan

$$(2t^2 + t)^2 - 4(2t^2 + t) + 3 = 0$$
$$\text{expand}((2t^2 + t)^2 - 4(2t^2 + t) + 3 = 0)$$
$$4t^4 + 4t^3 - 7t^2 - 4t + 3 = 0$$
$$\text{solve}((2t^2 + t)^2 - 4(2t^2 + t) + 3 = 0)$$
$$\left[ t = \frac{1}{2}, t = -1, t = 1, t = -\frac{3}{2} \right]$$

2) Selesaikan

$$x^4 - 8x^2 = 9$$
$$\text{solve}(x^4 - 8x^2 = 9, x)$$
$$[x = -i, x = i, x = -3, x = 3]$$

3) Selesaikan

$$(x - 2)^3 = x^3 - 2$$
$$\text{solve}((x - 2)^3 = x^3 - 2, x)$$
$$[x = 1]$$

4) Carilah solusi dari

$$x^2 - 2x = 15$$
$$\text{solve}(x^2 - 2x = 15)$$
$$[x = -3, x = 5]$$

5) Carilah solusi dari

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

$$\left[ x = \frac{1 - \sqrt{47}i}{8}, x = \frac{\sqrt{47}i + 1}{8} \right]$$

3.4

1) Selesaikan

$$\frac{1}{4} + \frac{1}{5} = \frac{1}{t}$$

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

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

2) Selesaikan

$$\frac{t+1}{3} - \frac{t-1}{2} = 1$$

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

$$[t = -1]$$

3) Selesaikan

$$\sqrt{2x+5} = x-5$$

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

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

4) Selesaikan

$$\sqrt{y+7} + \sqrt{y+16} = 9$$

$$\text{solve}(\text{sqrt}(y+7) + \text{sqrt}(y+16) = 9)$$

$$\left[ \sqrt{y+16} = 9 - \sqrt{y+7} \right]$$

5) Selesaikan

$$\sqrt{3x-4} = 1$$

$$\text{solve}(\text{sqrt}(3x-4) = 1)$$

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

3.5

1) Selesaikan

$$|x+3| - 2 = 8$$

$$\text{solve}(\text{abs}(x + 3) - 2 = 8, x); \text{solve}((x+3)=10)$$

$$[x = 7]$$

2) Selesaikan

$$|4x - 3| + 1 = 7$$

$$\text{solve}(\text{abs}(4x - 3) + 1 = 7, x); \text{solve}(4x-3=6)$$

$$\left[ x = \frac{9}{4} \right]$$

3) Selesaikan

$$7 - |2x - 1| = 6$$

$$\text{solve}(7 - \text{abs}(2x - 1) = 6); \text{solve}(2x-1=1)$$

$$[x = 1]$$

4) Selesaikan

$$|2x - 4| < -5$$

BAB 3

Selesaikan

a.

$$x + 5\sqrt{x} - 36 = 0$$

$$\text{solve}(x + 5\text{sqrt}(x) - 36 = 0)$$

$$[x = 36 - 5\sqrt{x}]$$

b.

$$\sqrt{x + 4} - 2 = 1$$

$$\text{solve}(\text{sqrt}(x + 4) - 2 = 1)$$

$$[x = 5]$$

c.

$$R = \sqrt{3np}$$

$$\text{solve}(R = \text{sqrt}(3np), n)$$

Answering "Is R positive, negative or zero?" with "positive"

$$\left[ n = \frac{R^2}{3p} \right]$$

d.

$$|x + 4| = 7$$

*solve*( $x + 4 = 7, x$ )

$$[x = 3]$$

e.

$$|x + 5| > 2$$

*load*(*fourier\_elim*)

C:/Program Files/Euler x64/maxima/share/maxima/5.35.1/share/*fourier\_elim/fourier\_elim.lisp*  
*fourier\_elim*[*abs*( $x + 5$ )  $> 2$ ],  $x$ )

*fourier\_elim* <sub>$|x+5|>2$</sub>

Space between commands expected! Found: x) (character 120) You can disable this in the Options menu. Error in: *amp;fourier\_elim*[*abs*( $x + 5$ )*gt*; 2],  $x$ )...

4.1

1) Cari 0 pada polinomial fungsi dan kalikan tiap persamaan berikut

a.

$$f(x) = (x^2 - 5x + 6)^2$$

*expand*(( $x^2 - 5x + 6$ )<sup>2</sup>)

$$x^4 - 10x^3 + 37x^2 - 60x + 36$$

$px = x^4 - 10x^3 + 37x^2 - 60x + 36$ ; *px*

$$x^4 - 10x^3 + 37x^2 - 60x + 36$$

*solve*(*px*, 1,  $y=0$ ), *px*(

Floating point error! secant:  $x_2 = x_1 - y_1 * (x_1 - x_0) / (y_1 - y_0)$ ; Try "trace errors"  
 to inspect local variables after errors. *solve*: if *eps* then return *secant*(*f*, *a*, *b*, *y*; *args*()), *eps* = *eps*);

### 3 Plot 2D

23030630036\_MarcellineCalyaPadmarini\_TUGASEMT2\_MarcellineCalyaPadmarini

23030630036

Matematika E 2023

Menggambar Grafik 2D dengan EMT

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

Plot Dasar

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

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

```
clg; // clear screen
window(0,0,1024,1024); // use all of the window
setplot(0,1,0,1); // set plot coordinates
hold on; // start overwrite mode
n=100; X=random(n,2); Y=random(n,2); // get random points
colors=rgb(random(n),random(n),random(n)); // get random colors
loop 1 to n; color(colors[]); plot(X[],Y[]); end; // plot
hold off; // end overwrite mode
insimg; // insert to notebook
![images/23030630036_Marcelline
reset;
```

Ini penting untuk menahan grafik, karna perintah `plot()` akan menghapus jendela plot.

Untuk menghapus semua yang kita buat, kami 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.

Untuk contoh lain, kita menggambar plot sisipan di plot lain. Hal ini dilakukan dengan mendefinisikan jendela plot yang lebih kecil. Perhatikan bahwa jendela ini tidak menyediakan ruang untuk label sumbu diluar jendela plot. Kita harus menambahkan beberapa batasan untuk ini sesuai dengan kebutuhan. Perhatikan bahwa kita menyimpan dan memulihkan jendela penuh, dan menahan plot saat ini saat kami memplot inset.

```
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/23030630036_Marcelline
hold off;
window(ow);
```

Sebuah plot dengan beberapa bentuk dengan cara yang sama. Ada fungsi `figure()` untuk utilitas ini.

Aspek Plot

Plot standar(default) menggunakan jendela persegi. Anda dapat mengubahnya dengan fungsi `aspek()`. Jangan lupa untuk mengatur ulang aspek setelahnya. Anda juga dapat mengubah standar ini pada menu dengan menggunakan "Atur Aspek" untuk aspek rasio tertentu atau ke ukuran jendela grafik saat ini.

Tetapi Anda juga dapat mengubahnya menjadi 1 plot. Untuk ini, ukuran area plot saat ini diubah dan jendela diatur sehingga memiliki cukup ruang.

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

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

```
!images/23030630036_Marcelline
```

```
aspek();
```

```
reset;
```

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

2D Plots in Euler

EMT Math Toolbox memiliki plot di 2D, baik untuk data dan fungsi. EMT menggunakan fungsi plot 2D. Fungsi ini dapat membuat plot fungsi dan data.

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

Euler dapat memplot plot 2D dari

- ekspresi
- fungsi, variabel, atau kurva parametrik,
- 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 di "x" (contoh " $4*x^2$ ") atau sebuah nama fungsi (contoh " $f$ ") menghasilkan grafik fungsi.

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

Catatan: Jika kamu mengakhiri baris perintah dengan titik dua ":", plot tersebut akan di-inputkan ke jendela teks. Jika tidak, tekan TAB untuk melihat plot jika jendela plot tertutup.

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

```
!images/23030630036_Marcelline
```

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

```
!images/23030630036_Marcelline
```

```
a:=5.6; plot2d("exp(-ax^2)/a"); insimg(30); //menampilkangambarhasilplotsetinggi25baris
```

```
!images/23030630036_Marcelline
```

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 berikut:

- a,b: rentang-x (default -2,2)
- c,d: rentang-y (default: skala dengan nilai)

- r: sebagai alternatif radius di sekitar pusat plot
- cx,cy: koordinat pusat plot(default 0,0)

```
plot2d("x3 - x", -1, 2) :
!images/23030630036_Marcelline
plot2d("sin(x)", -2pi, 2pi): // plot sin(x) pada interval [-2pi, 2pi]
!images/23030630036_Marcelline
plot2d("cos(x)", "sin(3x)", xmin=0, xmax=2pi):
!images/23030630036_Marcelline
```

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

Dalam opsi, plot dapat diatur untuk muncul

- di jendela terpisah yang mampu diatur,
- di jendela notebook.

Lebih banyak gaya dapat dicapai dengan perintah plot tertentu.

Bagaimanapun, tekan tombol tabulator untuk melihat plot, jika itu disembunyikan.

Untuk membagi jendela menjadi beberapa plot, gunakan fungsi figure(). Sebagai contoh, kita akan membuat plot  $x^4$  menjadi beberapa jendela. *figure(0)* mengatur ulang jendela default.

```
reset;
figure(2,2); ... for n=1 to 4; figure(n); plot2d("x" + n); end; ... figure(0) :
!images/23030630036_Marcelline
```

Pada plot2d(), terdapat gaya alternatif yang berfungsi dengan grid=x. Untuk gambaran umum, kami menunjukkan berbagai gaya kisi dalam satu gambar (lihat di bawah untuk perintah figure()). Gaya grid=0 tidak disertakan. Gaya 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/23030630036_Marcelline
```

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

Atau dengan alternatif, a, b, c, d dapat menjadi spesifik sebagai a=... dsb.

Dalam 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/23030630036_Marcelline
plot2d("sin(x)+cos(2x)", 0, 4pi):
!images/23030630036_Marcelline
```

Gambar yang dihasilkan dengan memasukkan plot ke jendela teks di direktori yang sama dengan buku catatan, secara default disubdirektori dengan nama "gambar". Mereka juga menggunakan ekspor HTML.

Anda dapat menandai gambar manapun dan menyalinnya ke clipboard dengan Ctrl+C. Tentu saja, anda juga dapat mengeksport grafik saat ini dengan fungsi yang ada di menu File.

Fungsi atau ekspresi di plot2d dievaluasi secara adaptif. Untuk kecepatan lebih, matikan plot adaptif dengan lt;adaptive dan angka spesifik dari subinterval dengan n=... Ini hanya diperlukan dalam kasus yang jarang terjadi.

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

```

!images/23030630036_Marcelline
plot2d("x^x", r = 1.2, cx = 1, cy = 1) :
!images/23030630036_Marcelline
Perhatikan bahwa  $x^x$  tidak didefinisikan untuk  $x = 0$ . Fungsi plot2d() menangkap error ini, dan mulai mengplot
plot2d("log(x)", -0.1, 2):
!images/23030630036_Marcelline
Parameter square=true (atau gt;square) memilih y-range secara otomatis
sehingga hasilnya adalah jendela
plot persegi. Perhatikan bahwa secara default, Euler menggunakan ruang
persegi di dalam jendela plot.
plot2d("x^3 - x", square) :
!images/23030630036_Marcelline
plot2d("integrate(sin(x)exp(-x^2), 0, x)", 0, 2) : //plotintegral
!images/23030630036_Marcelline
Jika Anda membutuhkan lebih banyak ruang untuk label-y, panggil shrinkwin-
dow() dengan parameter yang lebih kecil, atau tetapkan nilai positif untuk "lebih
kecil" di plot2d().
plot2d("gamma(x)", 1, 10, yl="y-values", smaller=6, ivertical):
!images/23030630036_Marcelline
Ekspresi simbolik dapat digunakan karena disimpan sebagai ekspresi string
sederhana.
x=linspace(0, 2pi, 1000); plot2d(sin(5x), cos(7x)):
!images/23030630036_Marcelline
a:=5.6; expr = exp(-ax^2)/a; //defineexpression
plot2d(expr, -2, 2): // plot from -2 to 2
!images/23030630036_Marcelline
plot2d(expr, r=1, thickness=2): // plot in a square around (0,0)
!images/23030630036_Marcelline
plot2d(diff(expr, x), add, style="-", color=red): // add another plot
!images/23030630036_Marcelline
plot2d(diff(expr, x, 2), a=-2, b=2, c=-2, d=1): // plot in rectangle
!images/23030630036_Marcelline
plot2d(diff(expr, x), a=-2, b=2, square): // keep plot square
!images/23030630036_Marcelline
plot2d("x^2", 0, 1, steps = 1, color = red, n = 10) :
!images/23030630036_Marcelline
plot2d("x^2", add, steps = 2, color = blue, n = 10) :
!images/23030630036_Marcelline
Fungsi Satu Parameter

```

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

Berikut adalah beberapa contoh dalam menggunakan fungsi. Seperti biasanya pada EMT, fungsi tersebut bekerja untuk fungsi atau ekspresi lain, anda dapat meneruskan parameter tambahan (selain  $x$ ) yang bukan variabel global untuk fungsi dengan parameter titik koma atau koleksi panggilan.



```

function f(x,a) := x2/a + ax2 - x; //define a function
a=0.3; plot2d("f",0,1;a): // plot with a=0.3
!images/23030630036_Marcelline
plot2d("f",0,1;0.4): // plot with a=0.4
!images/23030630036_Marcelline
plot2d("f",0.2,0,1): // plot with a=0.2
!images/23030630036_Marcelline
plot2d("f(x,b)",b=0.1,0,1): // plot with 0.1
!images/23030630036_Marcelline
function f(x) := x3 - x; ... plot2d("f",r = 1) :
!images/23030630036_Marcelline

```

Berikut adalah ringkasan dari fungsi yang diterima

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

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

```

function f(x) = diff(xx, x)
x x (log(x) + 1)
plot2d(f,0,2):
!images/23030630036_Marcelline

```

Tentu saja, untuk ekspresi atau ekspresi simbolik dengan nama variabel saja cukup untuk memplotnya

```

expr = sin(x)exp(-x)
- x E sin(x)
plot2d(expr,0,3pi):
!images/23030630036_Marcelline
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/23030630036_Marcelline

```

Untuk gaya garis ada berbagai pilihan.

- \* gaya="...". Dipilih dari "-", "\_", "-.", ".", ":", "-.-", "-.-".
- \* warna: lihat dibawah untuk warna
- \* ketebalan: standarnya adalah 1

Warna dapat dipilih satu dari warna yang ada, atau sebagai RGB.

- \* 0..15: adalah default indeks warna
- \* konstanta warna: putih, hitam, merah, hijau, biru, cyan, zaitun, \* abu-abu muda, abu-abu, abu-abu tua, oranye, hijau muda, turquoise, biru \* muda, oranye muda, kuning
- \* rgb(red,green,blue): parameter adalah real pada [0,1].

```

plot2d("exp(-x2)",r = 2,color = red,thickness = 3,style = " - -") :
!images/23030630036_Marcelline

```

Ini adalah warna standar yang didefinisikan pada EMT.

```

aspect(2); columnsplot(ones(1,16),lab=0:15,grid=0,color=0:15):
!images/23030630036_Marcelline

```

Namun anda dapat menggunakan warna apa saja

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

```
!images/23030630036_Marcelline
```

Menggambar Beberapa Kurva pada bidang koordinat yang sama

Plot lebih dari satu fungsi (multiple functions) ke satu jendela dapat dilakukan dengan beberapa cara. Salah satunya yaitu menggunakan `gt;add` untuk memanggil ke `plot2d` secara keseluruhan, namun panggilan pertama. Kita telah menggunakan fitur ini dalam contoh diatas.

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

```
!images/23030630036_Marcelline
```

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

```
!images/23030630036_Marcelline
```

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

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

```
!images/23030630036_Marcelline
```

Kita menambahkan titik persimpangan dengan label (di posisi "cl" untuk kiri tengah), dan memasukan hasilnya ke dalam notebook. Kami juga menambahkan judul ke plot.

```
plot2d(["cos(x)","x"],r=1.1,cx=0.5,cy=0.5, ... color=[black,blue],style=["-","."], ... grid=1);
```

```
x0=solve("cos(x)-x",1); ... plot2d(x0,x0, points, add,title="Intersection Demo"); ... label("cos(x) = x",x0,x0,pos="cl",offset=20):
```

```
!images/23030630036_Marcelline
```

Dalam demo berikut, kami memplot  $\text{sinc}(x) = \sin(x)/x$  dan ekspansi Taylor ke-8 dan ke-16. Kami menghitung ekspansi ini menggunakan Maxima secara ekspresi simbolik.

Plot ini dilakukan dalam perintah multi-baris dengan 3 panggilan ke `plot2d()`. Yang kedua dan ketiga memiliki set flag `gt;add`, yang membuat plot ini menggunakan rentang sebelumnya.

Kami menambahkan kotak label yang menjelaskan fungsi.

```
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/23030630036_Marcelline
```

Pada contoh berikut, kami menghasilkan Bernstein-Polinomial.

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

```
plot2d("(1-x)^10",0,1); //plot first function
for i=1 to 10; plot2d("bin(10,i)x^i(1-x)^(10-i)", add); end;
insimg;
!images/23030630036_Marcelline
```

Metode kedua menggunakan pasangan matriks nilai-x dan matriks nilai-y yang berukuran sama.

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

```
x=linspace(0,1,500);
n=10; k=(0:n)'; // n is row vector, k is column vector
y=bin(n,k)xk(1-x)(n-k); //y is a matrix then
plot2d(x,y):
!images/23030630036_Marcelline
```

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/23030630036_Marcelline
```

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

```
plot2d(["sin(x)","cos(x)"],0,2pi,color=4:5):
!images/23030630036_Marcelline
plot2d(["sin(x)","cos(x)"],0,2pi): // plot vector of expressions
!images/23030630036_Marcelline
```

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]
mxm2str(v) // get a vector of strings from the symbolic vector
(1-x)1010*(1-x)9*x45*(1-x)8*x2120*(1-x)7*x3210*(1-x)6*x4252*
(1-x)5*x5210*(1-x)4*x6120*(1-x)3*x745*(1-x)2*x810*(1-x)*x9x10
plot2d(mxm2str(v),0,1): // plot functions
!images/23030630036_Marcelline
```

Alternatif lain adalah dengan menggunakan bahasa matriks Euler.

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

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

```
n=(1:10)'; plot2d("xn",0,1,color=1:10):
!images/23030630036_Marcelline
```

Ekspresi dan fungsi satu baris dapat melihat variabel global.

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

Berhati-hatilah, untuk meletakkan semua parameter yang ditetapkan di akhir perintah `plot2d`. Dalam contoh kita meneruskan `a=5` ke 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/23030630036\_Marcelline

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

Dalam contoh berikut, kami menggunakan loop untuk memplot beberapa fungsi (lihat tutorial tentang pemrograman untuk loop)

plot2d("f", 1, -10, 10); ... for a=2:10; plot2d("f", a, add); end:

!images/23030630036\_Marcelline

Kami dapat mencapai hasil yang sama dengan cara berikut menggunakan bahasa matriks EMT. Setiap baris matriks f(x,a) adalah fungsi. Selain itu, kita dapat mengatur warna untuk setiap baris matriks. Klik dua kali pada fungsi getspectral() untuk penjelasannya.

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

!images/23030630036\_Marcelline

Label Teks

Dekorasi sederhana bisa terdiri atas

\* sebuah judul dengan title="..."

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

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

Perintah label akan memplot ke dalam plot saat ini pada koordinat plot(x,y).

Itu bisa mengambil argumen posisi.

plot2d("x<sup>3</sup> - x", -1, 2, title = "y = x<sup>3</sup> - x", yl = "y", xl = "x") :

!images/23030630036\_Marcelline

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/23030630036\_Marcelline

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

function f(x) = x<sup>2</sup>exp(-x<sup>2</sup>); ... 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/23030630036\_Marcelline

Kotak ditambatkan di kanan atas secara default, tetapi gt; kiri menambatkannya di kiri atas. Anda dapat memindahkannya ke tempat yang Anda suka. Posisi jangkar adalah sudut kanan atas kotak, dan angkanya adalah pecahan dari ukuran jendela grafik. Lebarinya otomatis.

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

Dalam 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/23030630036\_Marcelline

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

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

!images/23030630036\_Marcelline

Fitur yang serupa adalah fungsi `textbox()`.

Lebar secara default adalah lebar maksimal dari baris teks. Tetapi itu bisa diatur oleh pengguna.

```
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/23030630036\_Marcelline

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/23030630036\_Marcelline

Label pada sumbu x dan y dapat ditulis secara vertikal, begitu juga sumbunya.

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

!images/23030630036\_Marcelline

LaTeX

Anda juga dapat memplot rumus LaTeX jika anda telah mengunduh sistem LaTeX. Saya merekomendasikan MiKTeX. Jalur ke biner "latex" dan "dvi2png" harus berada di jalur sistem, atau Anda harus mengatur LaTeX pada opsi menu.

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

Dalam plot berikut, kita menggunakan LaTeX untuk label-x dan y, sebuah label, kotak label, dan judul dari 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
fracsin(phi)phi"), x = 0.8, y = 0.5); ... label(latex("
Phi", color = blue), 1, 0.4) :
```

!images/23030630036\_Marcelline

Seringkali, kami menginginkan spasi dan label teks non-konformal pada sumbu x. Kita dapat menggunakan `x axis()` dan `y axis()` seperti yang akan kita tunjukkan nanti.

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

ilnya (sebuah gambar dalam matriks RGB).

Dalam plot berikut, kita menggunakan LaTeX untuk label-x dan y, sebuah label, kotak label, dan judul dari plot.

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

![images/23030630036\_Marcelline

Tentu saja, fungsi juga 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
-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 akan dapat menggunakan

LaTeX jika Anda telah menginstal LaTeX dengan benar.

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

$x^2x$

$le0$

$endcases$ "), ...  $x = 0.7, y = 0.2$ ) :

Variable f not found! Use global variables or parameters for string evaluation.

Error in expression: f y0=f(x[1],args()); adaptive evalone : s = Try"trace errors" to inspect local variables after  
dw/n, dw/n<sup>2</sup>, dw/n, auto; args());

Interaksi Pengguna

Ketika memploting suatu fungsi atau ekspresi, parameter `gt`;user mengizinkan pengguna untuk memperbesar dan menggeser plot dengan kursor atau mouse. Pengguna dapat

- \* memperbesar dengan + atau -
- \* memindahkan plot dengan kursor
- \* memilih jendela plot dengan mouse
- \* mengulang tampilan dengan spasi
- \* keluar dengan return

Tombol spasi akan mengatur ulang plot ke jendela plot asli.

Ketika memplot data, flag `gt`;user akan menunggu penekanan tombol.

```
plot2d("x^3 - ax",a = 1, user,title = "Press any key!") :
![images/23030630036_Marcelline
plot2d("exp(x)sin(x)",user=true, ... title="+/- or cursor keys (return to
exit)");
```

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

Fungsi bawaan `mousedrag()` menunggu event mouse atau keyboard. Ini melaporkan mouse ke bawah, mouse dipindahkan 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. Sebagai contoh, kita interpolasi dalam 5 titik dengan polinomial. Fungsi harus diplot ke area plot 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; Perhatikan
    parameter titik koma di plot2d (d dan xp), yang diteruskan ke evaluasi fungsi
    interp(). Tanpa ini, kita harus menulis fungsi plotinterp() terlebih dahulu, men-
    gakses nilai secara global.
```

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

```
t=-1:0.5:1; dragpoints("plotf",t,random(size(t))-0.5);
!images/23030630036_Marcelline
```

Ada juga fungsi yang memplot fungsi lain tergantung 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 membutuhkan nama untuk parameter, nilai awal dan matriks rentang  $nx2$ , opsional baris judul.

Ada slider interaktif yang dapat diatur nilainya oleh pengguna. Fungsi `dragvalues()` menyediakan ini.

```
dragvalues("plotf",["a","b"],[-1,2],[[-2,2],[1,10]], ... heading="Drag these
values:",hcolor=black);
!images/23030630036_Marcelline
```

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

```
Fungsi dragvalues() menyediakan ini.
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; Sekarang kami mengizinkan derajat  $n$  bervariasi dari 0 hingga 20 dalam
20 pemberhentian. Hasil dragvalues() digunakan untuk memplot sketsa ini den-
gan  $n$ , 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);
!images/23030630036_Marcelline
```

Berikut ini adalah demonstrasi sederhana dari fungsi tersebut. Pengguna dapat menggambar di atas jendela plot, meninggalkan jejak poin.

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

#### Gaya Plot 2D

Secara default, EMT menghitung titik sumbu otomatis dan menambahkan label untuk setiap titik. Ini dapat diubah dengan parameter grid. Gaya default sumbu dan label yang dimodifikasi. Selain itu, label dan judul dapat ditambahkan secara manual. Untuk mengatur ulang gaya default, gunakan reset().

```
aspect();
figure(3,4); ... figure(1); plot2d("x3-x", grid = 0); ...//nograd, frameoraxis
figure(2); plot2d("x3 - x", grid = 1); ...//x - y - axis
figure(3); plot2d("x3 - x", grid = 2); ...//defaultticks
figure(4); plot2d("x3 - x", grid = 3); ...//x - y - axiswithlabelsinside
figure(5); plot2d("x3 - x", grid = 4); ...//noticks, onlylabels
figure(6); plot2d("x3 - x", grid = 5); ...//default, butnomargin
figure(7); plot2d("x3 - x", grid = 6); ...//axesonly
figure(8); plot2d("x3 - x", grid = 7); ...//axesonly, ticksataxis
figure(9); plot2d("x3 - x", grid = 8); ...//axesonly, finerticksataxis
figure(10); plot2d("x3 - x", grid = 9); ...//default, smallticksinside
figure(11); plot2d("x3 - x", grid = 10); ...//noticks, axesonly
figure(0):
```

![images/23030630036\_Marcelline

Parameter lt;frame mematikan frame, dan framecolor=blue mengatur frame ke warna biru.

Jika Anda ingin membuat sendiri, Anda dapat menggunakan style=0, dan tambahkan semuanya nanti.

```
aspect(1.5);
plot2d("x3 - x", grid = 0); //plot
frame; xgrid([-1,0,1]); ygrid(0); // add frame and grid
![images/23030630036_Marcelline
```

Untuk judul plot dan label sumbu, dapat dilihat pada 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/23030630036_Marcelline
```

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/23030630036_Marcelline
```

Teks pada plot dapat diatur dengan label(). Dalam contoh berikut, "lc" berarti tengah bawah. Ini mengatur 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/23030630036_Marcelline
Ada juga kotak teks.
plot2d(f(x),-1,1,-2,2); // function
plot2d(diff(f(x),x), add,style="-",color=red); // derivative
labelbox(["f","f'"],["-", "-"],[black,red]): // label box
!images/23030630036_Marcelline
plot2d(["exp(x)","1+x"],color=[black,blue],style=["-", "-."]):
!images/23030630036_Marcelline
gridstyle("-",color=gray,textcolor=gray,framecolor=gray); ... plot2d("x3-
x",grid = 1); ... settitle("y = x3 - x",color = black); ... label("x",2,0,pos =
"bc",color = gray); ... label("y",0,6,pos = "cl",color = gray); ... reset() :
!images/23030630036_Marcelline
Untuk kontrol lebih, sumbu x dan sumbu y dapat dilakukan secara manual.
Perintah fullwindow() memperluas jendela plot karena kita tidak lagi mem-
butuhkan tempat untuk label di luar jendela plot. Gunakan shrinkwindow()
atau reset() untuk mengatur 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/23030630036_Marcelline
Berikut adalah contoh lain, di mana string Unicode digunakan dan sumbu
di luar area plot.
aspect(1.5);
plot2d(["sin(x)","cos(x)"],0,2pi,color=[red,green],jgrid,iframe); ... xaxis(-
1.1,(0:2)pi,xt=["0","u"pi","u"2pi"],style="-",ticks,zero); ... xgrid((0:0.5:2) ×
pi,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/23030630036_Marcelline

```

#### Memploting Data 2D

Jika x dan y merupakan vektor data, data ini akan digunakan sebagai koordinat x dan y dari kurva. Pada kasus ini, a, b, c, dan d, atau radius r dapat ditentukan, atau jendela plot akan otomatis menyesuaikan pada data. Selain itu, gt;square dapat diatur untuk menjaga aspek rasio persegi.

Memplot ekspresi hanyalah singkatan untuk plot data. Untuk plot data, Anda membutuhkan satu atau lebih baris nilai x, dan satu atau lebih baris nilai y. Dari rentang dan nilai-x, suatu fungsi plot2d akan menghitung data yang akan diplot, secara default dengan evaluasi fungsi yang adaptif. Untuk plot titik gunakan "gt;points", untuk campuran garis dan titik gunakan "gt;addpoints".

Tetapi Anda dapat memasukkan data secara langsung.

\* Menggunakan vektor baris untuk x dan y satu fungsi.

\* Matriks untuk x dan y diplot baris demi baris.  
Ini adalah contoh dengan satu baris untuk x dan y.  
 $x = -10:0.1:10$ ;  $y = \exp(-x^2)x$ ; *plot2d(x, y)* :  
!images/23030630036\_Marcelline

Data dapat diplot sebagai titik. Gunakan *points=true* untuk ini. Plot mbek-  
erja seperti poligon namun hanya menggambar sudut-sudutnya.  
\* *style="..."*: dipilih dari "*l*", "*lt*", "*gt*", "*o*", "*.*", "*..*", "*+*", "*\**", "*\**" "*l*", "*lt*", "*gt*", "*o*", "*.*", "*..*", "*+*", "*\**", "*\**".  
Untuk memplot set poin gunakan *gt;points*. Jika warna adalah vektor warna,  
tiap titik mendapatkan warna yang berbeda. Untuk matriks koordinat dan  
vektor kolom, warna yang berlaku adalah pada baris matriks.  
Parameter *gt;addpoints* menambahkan titik pada segmen garis untuk plot  
data.  
 $xdata = [1, 1.5, 2.5, 3, 4]$ ;  $ydata = [3, 3.1, 2.8, 2.9, 2.7]$ ; // data  
*plot2d(xdata, ydata, a=0.5, b=4.5, c=2.5, d=3.5, style=".")*; // lines  
*plot2d(xdata, ydata, points, add, style="o")*; // add points  
!images/23030630036\_Marcelline  
 $p = \text{polyfit}(xdata, ydata, 1)$ ; // get regression line  
*plot2d("polyval(p, x)", add, color=red)*; // add plot of line  
!images/23030630036\_Marcelline

Menggambar Daerah Yang Dibatasi Kurva  
Plot data benar-benar poligon. Kita juga dapat memplot kurva atau kurva  
terisi.  
\* *filled=true* mengisi plot.  
\* *style="..."*: dipilih dari "*l*", "*/*", "*.*", "*.*".  
\* *fillcolor*: Lihat diatas untuk warna yang tersedia  
Warna didalam fungsi ditentukan oleh argumen "*fillcolor*", dan pada *lt;outline*  
opsional mencegah menggambar batas untuk semua gaya kecuali yang default.  
 $t = \text{linspace}(0, 2\pi, 1000)$ ; // 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/23030630036\_Marcelline

Dalam contoh berikut kami memplot elips terisi dan dua segi enam terisi  
menggunakan kurva tertutup dengan 6 titik dengan gaya isian berbeda.  
 $x = \text{linspace}(0, 2\pi, 1000)$ ; *plot2d(sin(x), cos(x)0.5, r=1, filled, style="/")*:  
!images/23030630036\_Marcelline  
 $t = \text{linspace}(0, 2\pi, 6)$ ; ... *plot2d(cos(t), sin(t), filled, style="/", fillcolor=red, r=1.2)*:  
!images/23030630036\_Marcelline  
 $t = \text{linspace}(0, 2\pi, 6)$ ; *plot2d(cos(t), sin(t), filled, style="")*:  
!images/23030630036\_Marcelline

Contoh lainnya adalah segi empat, yang kita buat dengan 7 titik pada  
lingkaran satuan.  
 $t = \text{linspace}(0, 2\pi, 7)$ ; ... *plot2d(cos(t), sin(t), r=1, filled, style="/", fillcolor=red)*:

!images/23030630036\_Marcelline

Berikut ini adalah himpunan nilai maksimal dari empat kondisi linier yang kurang dari atau sama dengan 3. Ini adalah  $A[k].vlt;=3$  untuk setiap 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/23030630036\_Marcelline

Poin utama dari bahasa matriks adalah memungkinkan untuk menghasilkan 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. Plotnya bisa diisi. Pada kasus ini, ini menghasilkan hasil yang bagus karena aturan lilitan, yang digunakan untuk isi.

```
plot2d(x,y,igrid,ifram,ifilled):
!images/23030630036_Marcelline
```

Sebuah vektor interval diplot terhadap nilai x sebagai daerah terisi antara nilai interval bawah dan atas.

Hal ini dapat berguna untuk memplot kesalahan perhitungan. Tapi itu bisa juga 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/23030630036_Marcelline
```

Jika x adalah vektor yang diurutkan, dan y adalah vektor interval, maka plot2d akan memplot rentang interval yang terisi dalam bidang. Gaya isian sama dengan gaya poligon.

```
t=-1:0.01:1; x= t-0.01,t+0.01 ; y=x^3 - x;
plot2d(t,y):
!images/23030630036_Marcelline
```

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 := "2x^2 + xy + 3y^4 + y"; //define an expression f(x,y)
plot2d(expr,level=[0;1],style="-",color=blue): // 0 i= f(x,y) i= 1
!images/23030630036_Marcelline
```

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/23030630036\_Marcelline

```
plot2d("cos(x)","sin(x)^3",xmin = 0,xmax = 2pi, filled,style = "/" ) :
!images/23030630036_Marcelline
```

Grafik Fungsi Parametrik

Nilai x tidak perlu diurutkan. (x,y) hanya mendefinisikan sebuah kurva. Jika x diurutkan, maka kurva tersebut merupakan fungsi grafik.

Dalam contoh berikut, kita memplot spiral

$$\gamma(t) = t \cdot (\cos(2\pi t), \sin(2\pi t))$$

Kita perlu menggunakan banyak titik untuk tampilan yang halus atau fungsi adaptif() untuk mengevaluasi eksresi (lihat fungsi adaptif() untuk lebih jelasnya).

```
t=linspace(0,1,1000); ... plot2d(tcos(2pit),tsin(2pit),r=1):
```

```
![images/23030630036_Marcelline
```

Atau, 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/23030630036_Marcelline
```

```
t=linspace(0,1,1000); r=exp(-t); x=rcos(2pit); y=rsin(2pit);
```

```
plot2d(x,y,r=1):
```

```
![images/23030630036_Marcelline
```

Pada contoh berikutnya, kita memplot kurva

$$\gamma(t) = (r(t) \cos(t), r(t) \sin(t))$$

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/23030630036_Marcelline
```

Menggambar Grafik Bilangan Kompleks

Array bilangan kompleks juga dapat diplot. Kemudian titik-titik grid akan terhubung. Jika sejumlah garis kisi ditentukan (atau vektor garis kisi 1x2) dalam argumen cgrid, hanya garis kisi tersebut yang terlihat.

Matriks bilangan kompleks akan secara otomatis diplot sebagai kisi di bidang kompleks.

Dalam contoh berikut, kami 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(Ia);... plot2d(z,a=-1.25,b=1.25,c=-1.25,d=1.25,cgrid=10):
```

```
![images/23030630036_Marcelline
```

```
aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)'; z=rexp(Ia);
```

```
plot2d(exp(z),cgrid=[40,10]):
```

```
![images/23030630036_Marcelline
```

```
r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=rexp(Ia);
```

```
plot2d(exp(z), points, add):
```

```
![images/23030630036_Marcelline
```

Sebuah vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian real dan bagian imajiner.

Dalam contoh, kami memplot lingkaran satuan dengan

$$\gamma(t) = e^{it}$$

```
t=linspace(0,2pi,1000); ... plot2d(exp(1t)+exp(4It),r=2):
!images/23030630036_Marcelline
Plot Statistik
```

Ada banyak fungsi yang dikhususkan pada plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

Jumlah kumulatif dari 0-1-distribusi normal menghasilkan jalan acak.

```
plot2d(cumsum(randnormal(1,1000))):
```

```
!images/23030630036_Marcelline
```

Menggunakan dua baris menunjukkan jalan dalam dua dimensi.

```
X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]):
```

```
!images/23030630036_Marcelline
```

```
columnplot(cumsum(random(10)),style="/",color=blue):
```

```
!images/23030630036_Marcelline
```

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];
```

```
columnplot(values,lab=months,color=red,style="-");
```

```
title("Temperature");
```

```
!images/23030630036_Marcelline
```

```
k=0:10;
```

```
plot2d(k,bin(10,k), bar):
```

```
!images/23030630036_Marcelline
```

```
plot2d(k,bin(10,k)); plot2d(k,bin(10,k), points, add):
```

```
!images/23030630036_Marcelline
```

```
plot2d(normal(1000),normal(1000), points,grid=6,style="..");
```

```
!images/23030630036_Marcelline
```

```
plot2d(normal(1,1000), distribution,style="O");
```

```
!images/23030630036_Marcelline
```

```
plot2d("qnormal",0,5;2.5,0.5, filled):
```

```
!images/23030630036_Marcelline
```

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/23030630036_Marcelline
```

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/23030630036_Marcelline
```

Fungsi `statplot()` mengatur gaya dengan kalimat sederhana.

```
statplot(1:10,cumsum(random(10)),"b");
```

!images/23030630036\_Marcelline

```
n=10; i=0:n; ... plot2d(i,bin(n,i)/2^n, a = 0, b = 10, c = 0, d = 0.3); ...plot2d(i, bin(n, i)/2^n, points = true, style = "ow", add = true, color = blue) :
```

!images/23030630036\_Marcelline

Selain itu, data dapat diplot sebagai batang. Dalam hal ini, `x` harus diurutkan dan satu elemen lebih panjang dari `y`. Bilah akan memanjang dari `x[i]` ke `x[i+1]` dengan nilai `y[i]`. Jika `x` memiliki ukuran yang sama dengan `y`, maka 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/23030630036\_Marcelline

Data untuk plot batang (`bar=1`) dan histogram (`histogram=1`) dapat dinyatakan 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;genap` ditentukan, nilai `xv` akan dihitung dalam interval bilangan bulat.

```
plot2d(normal(10000),distribution=50):
```

!images/23030630036\_Marcelline

```
k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m, bar):
```

!images/23030630036\_Marcelline

```
columnplot(m,k):
```

!images/23030630036\_Marcelline

```
plot2d(random(600)6,histogram=6):
```

!images/23030630036\_Marcelline

Untuk distribusi, ada parameter `distribution=n`, yang menghitung nilai secara otomatis dan mencetak distributif relatif dengan `n` sub-interval.

```
plot2d(normal(1,1000),distribution=10,style="
```

/"):

!images/23030630036\_Marcelline

Dengan parameter `even=true`, ini akan menggunakan interval integer.

```
plot2d(inrandom(1,1000,10),distribution=10,even=true):
```

!images/23030630036\_Marcelline

Perhatikan bahwa terdapat banyak plot statistik yang mungkin akan berguna. Lihatlah tutorial tentang statistik.

```
columnplot(getmultiplicities(1:6,inrandom(1,6000,6))):
```

!images/23030630036\_Marcelline

```
plot2d(normal(1,1000), distribution); ... plot2d("qnormal(x)",color=red,thickness=2, add):
```

!images/23030630036\_Marcelline

Ada juga banyak plot khusus untuk statistik. Boxplot menunjukkan kuartil dari distribusi ini dan banyak outlier. Menurut definisi, outlier dalam boxplot adalah data yang melebihi 1,5 kali kisaran 50tengah plot.

```
M=normal(5,1000); boxplot(quantiles(M)):
```

!images/23030630036\_Marcelline

Fungsi Implisit

Plot implisit menunjukkan garis level yang menyelesaikan  $f(x,y)=\text{level}$ , di mana "level" dapat berupa nilai tunggal atau vektor nilai. Jika  $\text{level}=\text{"auto"}$ , akan ada garis level  $n$ , yang akan menyebar antara fungsi minimum dan maksimum secara merata. Warna yang lebih gelap atau lebih terang dapat ditambahkan dengan  $\text{gt;hue}$  untuk menunjukkan nilai fungsi. Untuk fungsi implisit,  $xv$  harus berupa fungsi atau ekspresi dari parameter  $x$  dan  $y$ , atau, sebagai alternatif,  $xv$  dapat berupa matriks nilai.

Euler dapat menandai garis level

$$f(x, y) = c$$

dari berbagai fungsi.

Untuk menggambar himpunan  $f(x,y)=c$  untuk satu atau lebih konstanta  $c$ , Anda dapat menggunakan `plot2d()` dengan plot implisit didalam bidang. Parameter untuk  $c$  adalah  $\text{level}=c$ , dimana  $c$  dapat berubah vektor garis level. Selin itu, skema warna dapat digambar pada latar belakang untuk menunjukan nilai fungsi setiap titik dalam plot. Parameter " $n$ " menentukan kehalusan plot.

```
aspect(1.5);
plot2d("x^2 + y^2 - xy - x", r = 1.5, level = 0, contourcolor = red) :
!images/23030630036_Marcelline
expr := "2x^2 + xy + 3y^4 + y"; //define an expression f(x, y)
plot2d(expr, level=0): // Solutions of f(x,y)=0
!images/23030630036_Marcelline
plot2d(expr, level=0:0.5:20, hue, contourcolor=white, n=200): // nice
!images/23030630036_Marcelline
plot2d(expr, level=0:0.5:20, hue, spectral, n=200, grid=4): // nicer
!images/23030630036_Marcelline
```

Ini berfungsi pula untuk plot data. Tetapi Anda harus menentukan rentang untuk label sumbu.

```
x=-2:0.05:1; y=x'; z=expr(x,y);
plot2d(z, level=0, a=-1, b=2, c=-2, d=1, hue):
!images/23030630036_Marcelline
plot2d("x^3 - y^2", contour, hue, spectral) :
!images/23030630036_Marcelline
plot2d("x^3 - y^2", level = 0, contourwidth = 3, add, contourcolor = red) :
!images/23030630036_Marcelline
z=z+normal(size(z))0.2;
plot2d(z, level=0.5, a=-1, b=2, c=-2, d=1):
!images/23030630036_Marcelline
plot2d(expr, level=[0:0.2:5;0.05:0.2:5.05], color=lightgray):
!images/23030630036_Marcelline
plot2d("x^2 + y^3 + xy", level = 1, r = 4, n = 100) :
!images/23030630036_Marcelline
plot2d("x^2 + 2y^2 - xy", level = 0 : 0.1 : 10, n = 100, contourcolor =
white, hue) :
!images/23030630036_Marcelline
```

Ini juga memungkinkan 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/23030630036_Marcelline
```

Plot implisit juga dapat menunjukkan rentang level. Kemudian level harus berupa matriks 2xn dari interval level, dimana baris pertama terdiri atas awalan dan baris kedua menunjukkan akhir dari setiap interval. Atau, vektor baris sederhana dapat digunakan untuk level, dan parameter dl memperluas nilai level ke interval.

```
plot2d("x^4 + y^4", r = 1.5, level = [0; 1], color = blue, style = "/" ) :
```

```
!images/23030630036_Marcelline
```

```
plot2d("x^2 + y^3 + xy", level = [0, 2, 4; 1, 3, 5], style = "/" , r = 2, n = 100) :
```

```
!images/23030630036_Marcelline
```

```
plot2d("x^2 + y^3 + xy", level = -10 : 20, r = 2, style = " - ", dl = 0.1, n = 100) :
```

```
!images/23030630036_Marcelline
```

```
plot2d("sin(x)cos(y)", r=pi, hue, levels,n=100):
```

```
!images/23030630036_Marcelline
```

Dimungkinkan juga untuk menandai suatu wilayah

$$a \leq f(x, y) \leq b.$$

Ini dilakukan dengan menambahkan level dengan dua baris.

```
plot2d("(x^2+y^2-1)^3-x^2y^3", r = 1.3, ...style = "", color = red, < outline, ...level = [-2; 0], n = 100) :
```

```
!images/23030630036_Marcelline
```

Dimungkinkan pula untuk menentukan level spesifik tertentu. Sebagai contoh, 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/23030630036_Marcelline
```

```
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/pre; Tidak ada kotak atau sumbu disini. Se-
lain itu, kami menggunakan jendela penuh untuk plot.
```



Kita memanggil reset sebelum menguji plot ini untuk mengembalikan grafik default. Hal ini tidak perlu, jika Anda yakin plot Anda berkerja.

```
reset; starplot1(normal(1,10)+5,color=red,lab=1:10):
```

```
!images/23030630036_Marcelline
```

Terkadang, Anda ingin memplot sesuatu yang plot2d tidak dapat lakukan, tetapi hampir.

Pada fungsi berikut, kami melakukan plot impuls logaritmik. plot2d dapat membuat plot logaritmik, tetapi tidak dengan batang impulsif.

```
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 > Marikitauidengannilaidistribusieksponen
```

```
        aspect(1.5); x=1:10; y=-log(random(size(x)))200; ... logimpulseplot1(x,y):
```

```
!images/23030630036_Marcelline
```

Mari kita menganimasikan kurva 2D menggunakan plot langsung. Perintah plot(x,y) hanya memplot kurva ke jendela plot. setplot(a,b,c,d) mengatur jendela ini.

Fungsi wait(0) memaksa plot untuk muncul di jendela grafik. Jika tidak, menggambar ulang terjadi 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; en-
```

```
    dif; f=f+0.02; end; framecolor(c); linewidth(l); endfunction i/pre; Tekan tombol
```

manapun untuk menghentikan animasi berikut.

```
    animliss(2,3); // lihat hasilnya, jika sudah puas, tekan ENTER
```

Plot Logaritma

EMT menggunakan parameter "logplot" untuk skala logaritmik. Plot Logaritma dapat diplot menggunakan skala logaritma dalam y dengan logplot=1, maupun menggunakan skala logaritma dalam x dan y dengan logplot=2, atau dalam x dengan logplot=3.

- logplot=1: logaritma-y - logplot=2: logaritma-x-y - logplot=3: logaritma-x

```
plot2d("exp(x3 - x)x2n", 1, 5, logplot = 1) :
```

```
!images/23030630036_Marcelline
```

```
plot2d("exp(x+sin(x))", 0,100,logplot=1):
```

```
!images/23030630036_Marcelline
```

```
plot2d("exp(x+sin(x))", 10,100,logplot=2):
```

```
!images/23030630036_Marcelline
```

```
plot2d("gamma(x)", 1,10,logplot=1):
```

```
!images/23030630036_Marcelline
```

```
plot2d("log(x(2+sin(x/100)))", 10,1000,logplot=3):
```

```
!images/23030630036_Marcelline
```

Ini juga bekerja pada plot data.

```
x=10(1 : 20); y = x2 - x;
```

```
plot2d(x,y,logplot=2):
```

```
!images/23030630036_Marcelline
```

Rujukan Lengkap Fungsi plot2d()



"", "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

23030630036\_MarcellineCalyaPadmarini\_TugasPlot3D>Nama : MarcellineCalyaPadmarini  
 NIM : 23030630036  
 Kelas : Matematika E  
 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 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^2 + sin(y)", -5, 5, 0, 6pi) :
```

```
!images/23030630036_Marcelline
```

```
plot3d("x^2 + xsin(y)", -5, 5, 0, 6pi) :
```

```
!images/23030630036_Marcelline
```

Silakan lakukan modifikasi agar gambar "talang bergelombang" tersebut tidak lurus melainkan melengkung/melingkar, baik melingkar secara mendatar maupun melingkar turun/naik (seperti papan peluncur pada kolam renang. Temukan rumusnya.

Fungsi dua variabel

Untuk grafik fungsi, gunakan

- \* ekspresi sederhana di x dan y,
- \* nama dari fungsi dua variabel
- \* atau matriks data.

Defaultnya adalah grid kawat yang terisi dengan warna berbeda di kedua sisi. Perhatikan bahwa jumlah interval kisi default adalah 10, tetapi plot menggunakan jumlah persegi panjang default 40x40 untuk membuat permukaan. Ini dapat diubah.

\* n=40, n=[40,40]: jumlah garis kisi di setiap arah

\* kisi=10, kisi=[10,10]: jumlah garis kisi di setiap arah.

Kami menggunakan default n=40 dan grid=10.

```
plot3d("x^2 + y^2") :
```

```
!images/23030630036_Marcelline
```

Interaksi pengguna memungkinkan dengan menggunakan parameter gt;user.

Pengguna dapat menekan tombol berikut.

\* left,right,up,down: memutar sudut pandang

\* +,-: memperbesar atau memperkecil

\* a: menghasilkan anaglyph (lihat di bawah)

\* l: beralih memutar sumber cahaya (lihat dibawah)

\* space: menyetel ulang ke default

\* return: mengakhiri interaksi

```
plot3d("exp(-x^2+y^2)", user, ...title = "Turnwiththevectorkeys(pressreturntofinish)") :
```

!images/23030630036\_Marcelline

Rentang plot untuk fungsi dapat ditentukan dengan

- \* a,b: rentang x
- \* c,d: rentang y
- \* r: persegi simetris di sekitar (0,0).
- \* n: jumlah subintegral plot

Ada beberapa parameter untuk menskalakan fungsi atau mengubah tampilan grafik.

- fscale: skala ke nilai fungsi (defaultnya 1;fscale)
- frame: jenis bingkai (default 1)

plot3d("exp(-(x<sup>2</sup>+y<sup>2</sup>)/5)", r = 10, n = 80, fscale = 4, scale = 1.2, frame = 3, user) :

!images/23030630036\_Marcelline

Tampilan dapat diubah dengan berbagai cara.

- \* distance: jarak pandang ke plot.
- \* zoom: nilai zoom.
- \* angle: sudut terhadap sumbu y negatif dalam radian.
- \* height: ketinggian tampilan dalam radian.

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("x<sup>2</sup> + y", distance = 3, zoom = 1, angle = pi/2, height = 0) :

!images/23030630036\_Marcelline

Plot selalu terlihat ke pusat kubus plot. Anda dapat memindahkan pusat dengan parameter pusat.

plot3d("x<sup>4</sup> + y<sup>2</sup>", a = 0, b = 1, c = -1, d = 1, angle = -20, height = 20, ... center = [0.4, 0, 0], zoom = 5) :

!images/23030630036\_Marcelline

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(-x<sup>2</sup> - y<sup>2</sup>)", r = 2, < fscale, < scale, distance = 13, height = 50, ... center = [0, 0, -2], frame = 3) :

!images/23030630036\_Marcelline

Sebuah plot kutub juga tersedia. Parameter polar=true menggambar plot polar. Fungsi tersebut harus tetap merupakan fungsi dari x dan y. Parameter "fscale" menskalakan fungsi dengan skala sendiri. Jika tidak, fungsi diskalakan agar sesuai dengan kubus.

```

plot3d("1/(x^2 + y^2 + 1)", r = 5, polar, ... fscale = 2, hue, n = 100, zoom =
4, contour, color = blue) :
!images/23030630036_Marcelline
function f(r) := exp(-r/2)cos(r); ... plot3d("f(x^2 + y^2)", polar, scale =
[1, 1, 0.4], r = 2pi, frame = 3, zoom = 4) :
!images/23030630036_Marcelline
Rotasi parameter memutar fungsi dalam x di sekitar sumbu x.
* rotate=1: Menggunakan sumbu x
* rotate=2: Menggunakan sumbu z
plot3d("x^2 + 1", a = -1, b = 1, rotate = true, grid = 5) :
!images/23030630036_Marcelline
plot3d("x^2 + 1", a = -1, b = 1, rotate = 2, grid = 5) :
!images/23030630036_Marcelline
plot3d("sqrt(25-x^2)", a = 0, b = 5, rotate = 1) :
!images/23030630036_Marcelline
plot3d("xsin(x)", a=0, b=6pi, rotate=2):
!images/23030630036_Marcelline
Ini adalah plot dengan 3 fungsi.
plot3d("x", "x^2 + y^2", "y", r = 2, zoom = 3.5, frame = 3) :
!images/23030630036_Marcelline
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.
- level=... (atau level): Sebuah vektor nilai untuk garis kontur.

Standarnya adalah level="auto", yang menghitung beberapa garis level secara otomatis. Seperti yang Anda lihat di plot, level sebenarnya adalah rentang level.

Gaya default dapat diubah. Untuk plot kontur berikut, kami menggunakan grid yang lebih halus untuk 100x100 poin, skala fungsi dan plot, dan menggunakan sudut pandang yang berbeda.

```

plot3d("exp(-x^2-y^2)", r = 2, n = 100, level = "thin", ...contour, spectral, fscale =
1, scale = 1.1, angle = 45, height = 20) :
!images/23030630036_Marcelline
plot3d("exp(xy)", angle=100°, contour,color=green):
!images/23030630036_Marcelline

```

Bayangan default menggunakan warna abu-abu. Tetapi rentang warna spektral juga tersedia.

- gt; spektral: Menggunakan skema spektral default
- color=...: Menggunakan warna khusus atau skema spektral

Untuk plot berikut, kami menggunakan skema spektral default dan menambah jumlah titik untuk mendapatkan tampilan yang sangat halus.

```

plot3d("x^2 + y^2", spectral, contour, n = 100) :

```

!images/23030630036\_Marcelline

Alih-alih garis level otomatis, kita bisa mengatur nilai garis level. Ini akan menghasilkan garis level tipis alih-alih rentang level.

```
plot3d("x2 - y2", 0, 5, 0, 5, level = -1 : 0.1 : 1, color = redgreen) :
```

!images/23030630036\_Marcelline

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/23030630036\_Marcelline

Pada contoh berikut, kita memplot himpunan, dimana

$$f(x, y) = x^y - y^x = 0$$

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/23030630036\_Marcelline

Dimungkinkan untuk menunjukan bidang kontur di bawah plot. Warna dan jarak ke plot dapat ditentukan.

```
plot3d("x2 + y4", cp, cpcolor = green, cpdelta = 0.2) :
```

!images/23030630036\_Marcelline

Berikut adalah beberapa gaya lagi. Kami selalu mematikan frame, dan menggunakan berbagai skema warna untuk plot dan grid.

```
figure(2,2); ... 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/23030630036\_Marcelline

Ada beberapa skema spektral lain, bernomor 1 sampai 9. Tetapi Anda juga dapat menggunakan color=value, dimana nilai

- \* spectral: untuk rentang dari biru sampai merah

- \* white: untuk rentang yang lebih redup

- \* yellowblue, purplegreen, blueyellow, greenred

- \* blueyellow, greenpurple, yellowblue, redgreen

```
figure(3,3); ... for i=1:9; ... figure(i); plot3d("x2 + y2", spectral = i, contour, cp, < frame, zoom = 4); ... end; ...figure(0) :
```

!images/23030630036\_Marcelline

Sumber cahaya dapat diubah dengan l dan tombol kursor selama interaksi pengguna. Itu juga dapat diatur dengan parameter.

- \* light: arah untuk cahaya

- \* amb: cahaya sekitar antara 0 and 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/23030630036_Marcelline
```

Parameter warna dapat 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/23030630036_Marcelline
```

Warna 0 memberikan spesial efek pelangi.

```
plot3d("x2/(x2 + y2 + 1)",color = 0,hue = true,grid = 10) :
```

```
!images/23030630036_Marcelline
```

Permukaannya juga bisa transparan.

```
plot3d("x2 + y2", transparent, grid = 10, wirecolor = red) :
```

```
!images/23030630036_Marcelline
```

Plot Implisit

Ada juga plot implisit dalam tiga dimensi. Euler menghasilkan pemotongan melalui objek. Fitur plot3d termasuk plot implisit. Plot-plot ini menunjukkan himpunan nol dari suatu fungsi dalam tiga variabel. Solusi dari

$$f(x, y, z) = 0$$

dapat divisualisasikan dalam potongan sejajar dengan bidang x-y-, x-z- dan y-z.

\* implicit=1: potong sejajar dengan bidang y-z

\* implicit=2: potong sejajar dengan bidang x-z

\* implicit=4: potong sejajar dengan bidang x-y

Tambahkan nilai-nilai ini, jika Anda suka. Dalam contoh kita plot

$$M = \{(x, y, z) : x^2 + y^3 + zy = 1\}$$

```
plot3d("x2 + y3 + zy - 1",r = 5,implicit = 3) :
```

```
!images/23030630036_Marcelline
```

```
c=1; d=1;
```

```
plot3d("((x2 + y2 - c2)2 + (z2 - 1)2)((y2 + z2 - c2)2 + (x2 - 1)2)((z2 + x2 -
c2)2 + (y2 - 1)2) - d",r = 2,< frame, implicit, user) :
```

```
!images/23030630036_Marcelline
```

```
plot3d("x2 + y3 + zy - 1",r = 5,implicit = 3) :
```

```
!images/23030630036_Marcelline
```

```
plot3d("x2 + y2 + 4xz + z3", implicit, r = 2, zoom = 2.5) :
```

```
!images/23030630036_Marcelline
```

Memplot Data 3D

Sama seperti plot2d, plot3d menerima data. Untuk objek 3D, Anda perlu menyediakan matriks nilai x,y- dan z, atau tiga fungsi atau ekspresi fx(x,y), fy(x,y), fz(x,y).

$$\gamma(t, s) = (x(t, s), y(t, s), z(t, s))$$

Karena  $x, y, z$  adalah matriks, kita asumsikan bahwa  $(t, s)$  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 polar.

```
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,ifframe,grid=[10,20], ...
values=s,contourcolor=red,level=[90°-24°;90°-22°], ... scale=1.4,height=50°):
```

![images/23030630036\_Marcelline

Ini adalah contoh yang termasuk sebuah fungsi grafik.

```
t=-1:0.1:1; s=(-1:0.1:1)'; plot3d(t,s,ts,grid=10):
```

![images/23030630036\_Marcelline

Namun, kita dapat membuat semua macam permukaan. Dengan menggunakan fungsi

$$x = yz$$

```
plot3d(ts,t,s,angle=180°,grid=10):
```

![images/23030630036\_Marcelline

Dengan usaha lebih, kita dapat menghasilkan banyak permukaan.

Pada contoh berikut, kami membuat tampilan bayangan bola yang terdistorsi. Koordinat biasa dari 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 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/23030630036\_Marcelline

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/23030630036\_Marcelline

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/23030630036_Marcelline
t=linspace(0,4pi,1000); plot3d(cos(t),sin(t),t/2pi, wire, ... linewidth=3, wirecolor=blue):
!images/23030630036_Marcelline
X=cumsum(normal(3,100)); ... plot3d(X[1],X[2],X[3], anaglyph, wire):
!images/23030630036_Marcelline

```

EMT juga dapat memplot mode anaglyph. Untuk melihat plot, Anda membutuhkan kacamata merah/cyan.

```

plot3d("x2 + y3", anaglyph, contour, angle = 30) :
!images/23030630036_Marcelline

```

Seringkali, skema warna spektral digunakan untuk plot. Ini menekankan ketinggian fungsi

```

plot3d("x2y3 - y", spectral, contour, zoom = 3.2) :
!images/23030630036_Marcelline

```

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/23030630036_Marcelline

```

Ini adalah contoh lebih yang lebih komplikasi, dimana hanya dapat dilihat dengan kacamata merah/cyan.

```

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/23030630036_Marcelline

```

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/23030630036_Marcelline
```

Dimungkinkan untuk membagi plot permukaan menjadi dua atau lebih bagian.

- \* x: vektor baris dengan n+1 elemen
- \* y: vektor kolom dengan n+1 elemen
- \* z: matriks nilai nxn.

z bisa lebih besar, tetapi hanya nilai nxn yang akan digunakan.

Dalam contoh, pertama-tama kita menghitung nilainya. Kemudian kita sesuaikan x dan y, sehingga vektor berpusat pada nilai yang digunakan.

```

x=-1:0.1:1; y=x'; z=x+y; d=zeros(size(x)); ... plot3d(x,y,z,disconnect=2:2:20):
!images/23030630036_Marcelline
Jika memuat atau menghasilkan matriks data M dari file dan perlu mem-
plotnya 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(ij^2+100*normal(20,20), zscale, scale = [1, 1, 1.5], angle =
-40, zoom = 1.8) :
!images/23030630036_Marcelline
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/23030630036_Marcelline
Permukaan Benda Putar
plot2d("(x^2+y^2-1)^3-x^2*y^3", r = 1.3, ...style = "", color = red, < outline, ...level =
[-2; 0], n = 100) :
!images/23030630036_Marcelline
plot2d("(x^2 + y^2 - 1)^3 - x^2*y^3", r = 1.3, ... style = "", color = blue, <
outline, ... level = [-4; 0], n = 120) :
!images/23030630036_Marcelline
plot2d("(x^2+y^2-1)^3-x^2*y^3", r = 1.3, ...style = "", color = red, < outline, ...level =
[-2; 0], n = 150) :
!images/23030630036_Marcelline
ekspresi = (x^2 + y^2 - 1)^3 - x^2*y^3;ekspresi

```

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

Kami ingin memmutar kurva hati disekitar sumbu-y. Berikut adalah ekspresi yang mendefinisikan hati:

$$f(x, y) = (x^2 + y^2 - 1)^3 - x^2 y^3.$$

kemudian himpunan

$$x = r \cos(a), \quad y = r \sin(a).$$

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 hati 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/23030630036_Marcelline

```

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, 100, 60], anaglyph) :
!images/23030630036_Marcelline
```

Plot 3D Khusus

Fungsi plot3d bagus untuk dimiliki, tetapi tidak memenuhi semua kebutuhan. Selain rutinitas yang lebih mendasar, dimungkinkan untuk mendapatkan plot berbingkai dari objek apa pun yang Anda suka.

Meskipun Euler bukan program 3D, ia dapat menggabungkan beberapa objek dasar. Kami mencoba memvisualisasikan paraboloid dan garis singgungnya.

```
function myplot ...
    y=-1:0.01:1; x=(-1:0.01:1)'; plot3d(x,y,0.2*(x-0.1)/2,jscale,iframe,ihue, .. hues=0.5,icontour,color=orange);
h=holding(1); plot3d(x,y,(x2+y2)/2, < scale, < frame, > contour, > hue); holding(h); endfunction <
/pre > Sekarang framedplot() menyediakan framedanmengaturtampilan.
    framedplot("myplot",[-1,1,-1,1,0,1],height=0,angle=-30°, ... center=[0,0,-
0.7],zoom=3):
!images/23030630036_Marcelline
```

Dengan cara yang sama, Anda dapat memplot bidang kontur secara manual. Perhatikan bahwa plot3d() menyetel jendela ke fullwindow() secara default, tetapi plotcontourplane() mengasumsikan itu.

pa pun yang Anda suka.

Meskipun Euler bukan program 3D, ia dapat menggabungkan beberapa objek dasar. Kami mencoba memvisualisasikan paraboloid dan garis singgungnya.

```
x=-1:0.02:1.1; y=x'; z=x2-y4;
function myplot (x,y,z) ...    ipre class="udf" i zoom(2); wi=fullwindow();
plotcontourplane(x,y,z,level="auto",jscale); plot3d(x,y,z,ihue,jscale,iadd,color=white,level="thin");
window(wi); reset(); endfunction i/pre i myplot(x,y,z):
!images/23030630036_Marcelline
function testplot () := plot3d("x2 + y3"); ... rotate("testplot"); testplot() :
!images/23030630036_Marcelline
```

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() :
```

Press space to stop, return to end Press space to stop, return to end

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 subdirektori "bin" dari Povray ke jalur lingkungan, atau mengatur variabel "defaultpovray" dengan path lengkap yang menunjuk ke "pvengine.exe". <http://www.povray.org> dan meletakkan subdirektori "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 berada 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

```
plot3d("x^2 + y^2", zoom = 2) :
!images/23030630036_Marcelline
pov3d("x^2 + y^2", zoom = 3);
!images/23030630036_Marcelline
```

Kita dapat membuat fungsi transparan dan menambah penyelesaian lain.

Kita juga dapat menambah tingkat garis pada fungsi plot.

```
pov3d("x^2+y^3", axiscolor = red, angle = -45, anaglyph, ...look = povlook(cyan, 0.2), level =
-1 : 0.5 : 1, zoom = 3.8);
!images/23030630036_Marcelline
```

Terkadang perlu untuk mencegah penskalaan fungsi, dan menskalakan fungsi dengan tangan.

Kami memplot himpunan titik di bidang kompleks, di mana produk dari jarak ke 1 dan -1 sama dengan 1.

```
pov3d("((x-1)^2 + y^2)((x+1)^2 + y^2)/40", r = 2, ... angle = -120, level =
1/40, dlevel = 0.005, light = [-1, 1, 1], height = 10, n = 50, ... < fscale, zoom =
3.8);
!images/23030630036_Marcelline
```

Memploting dengan Koordinat

Alih-alih menggunakan fungsi, kita dapat memplot dengan suatu koordinat.

Sama seperti plot3d, kita membutuhkan 3 matriks untuk mendefinisikan objek.

Sebagai contoh, kita memutar fungsi di sekitar sumbu-z.

```
function f(x) := x^3 - x + 1; ... x = -1 : 0.01 : 1; t = linspace(0, 2pi, 50)'; ... Z =
x; X = cos(t)f(x); Y = sin(t)f(x); ... pov3d(X, Y, Z, angle = 40, height = 50, axis =
0, zoom = 4, light = [10, -5, 5]);
!images/23030630036_Marcelline
```

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=6,axis=0,height=30°,add=povsphere([0.5,0,0.25],0.15,povlook(r
... w=500,h=300);
!images/23030630036_Marcelline
```

Dengan metode bayangan canggi 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 permukannya adalah  $[x, y, Z]$ . Kami menghitung dua turunan ke  $x$  dan  $y$ , serta 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,
[0, 0, 1]
Kita hanya menggunakan 25 titik
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);
Unexpected "(". Index () not allowed in strict mode! In Euler files, use relax
to avoid this. Error in: pov3d(x,y,Z(x,y),angle=10°; xv=NX(x,y),yv=NY(x,y),zv=NZ(x,y)
```

... Berikut ini adalah simpul Trefoil yang dilakukan oleh A. Busser di Povray. Ada versi yang ditingkatkan dari ini dalam contoh.

```
ja href="ExamplesKnot.html";Trefoil Knot;/a;
Untuk tampilan yang bagus dengan tidak terlalu banyak titik, kami menam-
bahkan 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 diturunkan 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 dan[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.

normal di sini. Kami menggunakan Maxima untuk menghitung normal bagi kami. Pertama, ketiga fungsi koordinat sebagai ekspresi simbolik.

```
pov3d(X(x,y),Y(x,y),Z(x,y), anaglyph,axis=0,zoom=5,w=450,h=350, ...
;shadow,look=povlook(blue), ... xv="dn[1]"(x,y), yv="dn[2]"(x,y), zv="dn[3]"(x,y));
!images/23030630036_Marcelline
```

Kita juga dapat membuat sebuah grid di 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/23030630036_Marcelline
```

Dengan menggunakan 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/23030630036_Marcelline
```



Objek Povray

Di atas, kami menggunakan pov3d untuk memplot suatu 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 mulai keluaran dengan povstart().

```
povstart(zoom=4);
```

Pertama kita mendefinisikan tiga silinder, dan menyimpannya dalam string di Euler.

Fungsi povx() dll. hanya mengembalikan vektor [1,0,0], yang dapat digunakan sebagai gantinya.

```
c1=povcylinder(-povx,povx,1,povlook(red)); ... c2=povcylinder(-povy,povy,1,povlook(yellow));
... c3=povcylinder(-povz,povz,1,povlook(blue)); ... String berisi kode
Povray, yang tidak perlu kita pahami pada saat itu.
```

```
c2
cylinder lt;0,0,-1gt;; lt;0,0,1gt;; 1 texture pigment color rgb lt;0.941176,0.941176,0.392157gt;
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/23030630036_Marcelline
```

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,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/23030630036_Marcelline
```


```

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 pada povobject(), yang mengembalikan string seperti biasanya.

```
c1=povobject("mycube",povlook(red));
```

Kami menghasilkan kubus kedua dan memutar serta menskalakannya sedikit.

```
c2=povobject("mycube",povlook(yellow),translate=[1,1,1], ... rotate=xrotate(10°)+yrotate(10°),
scale=1.2);
```

Kemudian kami mengambil perbedaan dari kedua objek.

```
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/23030630036\_Marcelline

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.

$$((x^2+y^2-c^2)^2+(z^2-1)^2)*((y^2+z^2-c^2)^2+(x^2-1)^2)*((z^2+x^2-c^2)^2+(y^2-1)^2) = d$$

```
povstart(angle=70°,height=50°,zoom=4);
```

```
writeAxes(); ... writeln(povsurface("(pow(pow(x,2)+pow(y,2)-pow(c,2),2)+pow(pow(z,2)-
1,2))(pow(pow(y,2)+pow(z,2)-pow(c,2),2)+pow(pow(x,2)-1,2))(pow(pow(z,2)+pow(x,2)-
pow(c,2),2)+pow(pow(y,2)-1,2))-d",povlook(red))); ... povend();
```

Error : Povray error!

Error generated by error() command

```
povray: error("Povray error!"); Try "trace errors" to inspect local variables
after errors. povend: povray(file,w,h,aspect,exit);
```

```
povstart(angle=25°,height=10°);
```

```
writeln(povsurface("pow(x,2)+pow(y,2)pow(z,2)-1",povlook(blue),povbox(-
2,2,"")));
```

```
povend();
```

!images/23030630036\_Marcelline

```
povstart(angle=70°,height=50°,zoom=4);
```

Buat permukaan implisit. Perhatikan perbedaan sintaks pada ekspresi.

```
writeln(povsurface("pow(x,2)y-pow(y,3)-pow(z,2)",povlook(green))); ... writeAxes();
... povend();
```

```

object isosurface function pow(x,2)*y-pow(y,3)-pow(z,2) max_gradient5opencontained,yboxlt;-1,-1,-1;
! [images/23030630036_Marcelline

```

Objek Jala

Pada contoh ini, kami menunjukkan bagaimana membuat objek jala, dan menggambarinya dengan informasi tambahan.

Kami ingin memaksimalkan xy dibawah kondisi  $x+y=1$  dan menunjukkan sentuhan tangen 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 mendefinisikan dua cakram, yang akan berpotongan dengan permukaan.

```
cl=povdisc([0.5,0.5,0],[1,1,0],2); ... ll=povdisc([0,0,1/4],[0,0,1],2);
```

Tulis permukaan dikurangi dua cakram.

```
writeln(povdifference(mesh,povunion([cl,ll]),povlook(green)));
```

Tulis dua persimpangan

```
writeln(povintersection([mesh,cl],povlook(red))); ... writeln(povintersection([mesh,ll],povlook(gray)));
```

Tulis titik 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/23030630036_Marcelline
```

Anaglyphs di Povray

Untuk menghasilkan sebuah anaglyph untuk kacamata merah/cyan, Povray harus berjalan dua kali dari posisi yang berbeda. Ini menghasilkan dua file Povray dan dua file PNG yang dimuat dalam suatu fungsi loadanaglyph().

Tentu saja, anda membutuhkan kacamata merah/cyan untuk melihat contoh berikut dengan benar.

Fungsi pov3d() memiliki sakelar sederhana untuk menghasilkan anaglyphs.

```
pov3d("exp(-x2-y2)/2",r=2,height=45,anaglyph,...center=[0,0,0.5],zoom=3.5);
```

```
! [images/23030630036_Marcelline
```

Jika Anda membuat adegan dengan objek, Anda perlu menempatkan generasi adegan ke dalam fungsi, dan menjalankannya dua kali dengan nilai yang berbeda untuk parameter anaglyph

```
function myscene ...
```

```
s=povsphere(povc,1); cl=povcylinder(-povz,povz,0.5); clx=povobject(cl,rotate=xrotate(90°));
```

```
cly=povobject(cl,rotate=yrotate(90°)); c=povbox([-1,-1,0],1); un=povunion([cl,clx,cly,c]);
```

```
obj=povdifference(s,un,povlook(red)); writeln(obj); writeAxes(); endfunction
```

}; Fungsi povanaglyph() melakukan semua ini. Parameternya seperti di povstart() dan povend() digabungkan.

lankannya dua kali dengan nilai yang berbeda untuk parameter anaglyph

```
povanaglyph("myscene",zoom=4.5);
```

```
![images/23030630036_Marcelline
```

Mendefinisikan Objek Sendiri

Antarmuka povray Euler berisi banyak objek. Tapi Anda tidak terbatas pada ini. Anda dapat membuat objek sendiri, yang menggabungkan objek lain, atau objek yang sama sekali baru.

Kami mendemonstrasikan sebuah torus. Perintah Povray untuk ini adalah "torus". Jadi kami mengembalikan string dengan perintah ini dan parameternya. Perhatikan bahwa torus selalu berpusat di titik asal.

```
function povdonat (r1,r2,look="") ...
```

```
return "torus "+r1+" "+r2+look+""; endfunction ;/pre>; inilah torus pertama kami
```

```
t1=povdonat(0.8,0.2)
```

```
torus 0.8,0.2
```

Mari kita gunakan objek ini untuk membuat torus kedua, diterjemahkan dan diputar.

```
t2=povobject(t1,rotate=xrotate(90°),translate=[0.8,0,0])
```

```
object torus 0.8,0.2 rotate 90 *x translate lt;0.8,0,0gt;
```

Sekarang kita menempatkan objek-objek ini ke dalam sebuah adegan. Untuk tampilan, kami menggunakan Phong Shading.

```
povstart(center=[0.4,0,0],angle=0°,zoom=3.8,aspect=1.5); ... writeln(povobject(t1,povlook(green,phong=
```

```
... writeln(povobject(t2,povlook(green,phong=1))); ... gt;povend();
```

memanggil program Povray. Namun, dalam kasus error, ini tidak menunjukkan letak kesalahan. Karena itu Anda harus menggunakan

```
gt;povend(lt;exit);
```

jika ada yang tidak berhasil. Ini akan membiarkan jendela Povray terbuka.

```
povend(h=320,w=480);
```

```
![images/23030630036_Marcelline
```

Ini adalah contoh yang lebih rumit. Kita selesaikan

$$Ax \leq b, \quad x \geq 0, \quad c \cdot x \rightarrow \text{Max}.$$

dan tunjukkan titik layak serta optimal dalam plot 3D.

```
A=[10,8,4;5,6,8;6,3,2;9,5,6];
```

```
b=[10,10,10,10]';
```

```
c=[1,1,1];
```

Pertama, mari kita cek, jika contoh ini memiliki solusi.

```
x=simplex(A,b,c,max,check)'
```

```
[0, 1, 0.5]
```

Ya, sudah.

Kemudian kita definisikan dua objek. Yang pertama adalah

$$a \cdot x \leq b$$

```

function oneplane (a,b,look="") ...
    return povplane(a,b,look) endfunction i/pre;
Kemudian kita mendefinisikan
persimpangan dari semua setengah ruang dan sebuah kubus.
    function adm (A, b, r, look="") ...
        ol=[]; loop 1 to rows(A); ol=ol—oneplane(A[],b[]); end; ol=ol—povbox([0,0,0],[r,r,r]);
return povintersection(ol,look); endfunction i/pre;
Kita sekarang dapat meren-
canakan adegannya.
    povstart(angle=120°,center=[0.5,0.5,0.5],zoom=3.5); ...   writeln(adm(A,b,2,povlook(green,0.4)));
...   writeAxes(0,1.3,0,1.6,0,1.5); ...   Berikut ini adalah lingkaran di sekitar
optimal.
    writeln(povintersection([povsphere(x,0.5),povplane(c,c.x')], ...   povlook(red,0.9)));
Dan sebuah kesalahan di direktori optimum.
    writeln(povarrow(x,c0.5,povlook(red)));
Kami menambahkan teks ke layar. Teks hanyalah objek 3D. Kita perlu
menempatkan dan memutarinya menurut pandangan kita.
    tunjukan titik layak serta optimal dalam plot 3D.
    writeln(povtext("Linear Problem",[0,0.2,1.3],size=0.05,rotate=5°)); ...   povend();
!images/23030630036_Marcelline
Contoh Lain
Anda dapat mencari beberapa contoh lain dari Povray di Euler pada file
berikut.
ja href="Examples/Dandelin Spheres.html" iExamples/Dandelin Spheresi/a;
ja href="Examples/Donat Math.html" iExamples/Donat Mathi/a;
ja href="Examples/Trefoil Knot.html" iExamples/Trefoil Knoti/a;
ja href="Examples/Optimization by Affine Scaling.html" iExamples/Optimization
by Affine Scalingi/a;
load affinescaling;
type affinescaling
function affinescaling (A: real, b: column real, c: vector real, .. gamma:
number positive, x: none column real, history, infinite .. )
    Default for gamma : 0.9 Default for x : none Default for history : 0 Default
for infinite : 450359962737
    n=cols(A); if x==none then u=b-sum(A); A=A—u; x=ones(cols(A),1); c=c—infinite;
endif; z=c.x; if history then X=x; endif; repeat d=ones(1,cols(A)); d=x'; tA=A*d;
tc=c*d; y=fit(tA',tc'); v=tc'-tA'.y; while !all(v=0); if all(vlt;=0) then er-
ror("Problem unbounded"); endif; i=nonzeros(vgt;0); lambda=gamma/max(v[i]');
tx=x/d'-lambda*v; x=d'*tx; znew=c.x; if history then X=X—x; endif; while
znewlt;z; z=znew; end if history then return x[1:n],X[1:n]; else return x[1:n];
endif; endfunction
    xopt,X=affinescaling(A,b,c,gamma=0.5,x=[1,1,1]', history); xopt,
1 1 1
A=[1,1;4,5]—id(2)
1 1 1 0 4 5 0 1
b=[1000;4500]
1000 4500
c=[5,6,0,0]

```

```

[-5, -6, 0, 0]
simplex(A,b,c,eq=0, min, check)
500 500 0 0
affinescaling(A,b,c)
499.989 500.01 4.55637e-08 6.91608e-08

```

## 5 Kalkulus

23030630036\_MarcellineCalyaPadmarini\_EMT Kalkulus Nama : MarcellineCalyaPadmarini

Prodi : Matematika E

NIM : 23030630036

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\_fungsiamp; = rumus\_fungsi(untuk\_fungsi \* simbolik, namundapatdihitungsecaranumerik),

\* Menggunakan format nama\_fungsiamp; amp; = 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:

$$f(x) = 2x^2 + e^{\sin(x)}.$$

```
function f(x) := 2x^2 + exp(sin(x))/fungsiumerik
```

```
f(0), f(1), f(pi)
```

```
1 4.31977682472 20.7392088022
```

```
f(e) // tidak dapat dihitung nilainya
```

```
Variable or function e not found. Error in: f(e) // tidak dapat dihitung nilainya ...
```

Silakan Anda plot kurva fungsi di atas!

```
plot2d("f(x)");
```

```
![images/23030630036_Marcelline
```

—

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 vari-
ables after errors. g: useglobal; return sqrt(x^2 - 3 * x)/(x + 1)Error in :
g(1)//kompleks,tidakdapatdihitungolehfungsiumerik...
Silakan Anda plot kurva fungsi di atas!
plot2d("g(x)":
![images/23030630036_Marcelline
—
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
reset;
Silakan Anda plot kurva fungsi komposisi fungsi f dan g:
```

$$h(x) = f(g(x))$$

dan

$$u(x) = g(f(x))$$

bersama-sama kurva fungsi f dan g dalam satu bidang koordinat.

```
function h(x) := f(g(x))
function u(x) := g(f(x))
reset;
plot2d("h(x)",-20,20,-10,10):
![images/23030630036_Marcelline
plot2d("u(x)",-20,20):
![images/23030630036_Marcelline
plot2d("h(x)",-20,20,-10,10,color=red); plot2d("u(x)",color=blue,style="-", add):
![images/23030630036_Marcelline
f(0:10) // nilai-nilai f(0), f(1), f(2), ..., f(10)
[1, 4.31978, 10.4826, 19.1516, 32.4692, 50.3833, 72.7562, 99.929, 130.69,
163.51, 200.58]
```

```
fmap(0:10) // sama dengan f(0:10), berlaku untuk semua fungsi
[1, 4.31978, 10.4826, 19.1516, 32.4692, 50.3833, 72.7562, 99.929, 130.69,
163.51, 200.58]
gmap(200:210)
[0.987534, 0.987596, 0.987657, 0.987718, 0.987778, 0.987837, 0.987896, 0.987954,
0.988012, 0.988069, 0.988126]
```

Misalkan kita akan mendefinisikan fungsi

$$f(x) = \begin{cases} x^3 & x > 0 \\ x^2 & x \leq 0. \end{cases}$$

Fungsi tersebut tidak dapat didefinisikan sebagai fungsi numerik secara "inline" menggunakan format `:=`, melainkan didefinisikan sebagai program. Perhatikan, kata "map" digunakan agar fungsi dapat menerima vektor sebagai input, dan hasilnya berupa vektor. Jika tanpa kata "map" fungsinya hanya dapat menerima input satu nilai.

```
function map f(x) ...
if x<0 then return x^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/23030630036_Marcelline
function f(x) = 2E^x // fungsisimbolik
x 2 E
f(a) // nilai fungsisecarasimbolik
```

$$2e^a$$

```
f(E) // nilai fungsi berupa bilangan desimal
30.308524483
f(E),float(
```

30.30852448295852

```
![images/23030630036_Marcelline
function g(x) = 3x+1
3 x + 1
function h(x) = f(g(x)) // komposisi fungsi
3 x + 1 2 E
plot2d("h(x)",-1,1):
![images/23030630036_Marcelline
Latihan
```

Bukalah buku Kalkulus. Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan fungsi-fungsi tersebut dan komposisinya di EMT pada baris-baris perintah berikut



(jika perlu tambahkan lagi). Untuk setiap fungsi, hitung beberapa nilainya, baik untuk satu nilai maupun vektor. Gambar grafik fungsi-fungsi tersebut dan komposisi-komposisi 2 fungsi.

Juga, carilah fungsi beberapa (dua) variabel. Lakukan hal sama seperti di atas.

```

function f(x) := (x^2 - 4)/(x - 2) // fungsi numerik
f(1), f(0), f(-1)
3 2 1
f(2) // akan error karena pembagi bernilai 0
Floating point error! Try "trace errors" to inspect local variables after errors.
f: use global; return (x^2-4)/(x-2) Error in : f(2) // akan error karena pembagi bernilai 0...
f(3:15) // nilai f(3), f(4), ... f(15)
[5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17]
function g(x) := exp(sin(x)) // fungsi trigonometrik
g(pi), g(3)
1 1.15156283651
g(pi:4pi)
[1, 0.431076, 0.402807, 0.868385, 2.13145, 2.60889, 1.32236, 0.518411, 0.371815,
0.662246]
gmap(pi:4pi)
[1, 0.431076, 0.402807, 0.868385, 2.13145, 2.60889, 1.32236, 0.518411, 0.371815,
0.662246]
f(g(pi)), g(f(4))
3 0.756225627543
aspect(3); plot2d("f(g(x))", -5, 5):
! [images/23030630036_Marcelline
function f(x) := x^2
function g(x) := sqrt(x+4)
function p(x) := g(f(x)) // fungsi komposisi
p(5), g(f(5))
5.38516480713 5.38516480713
aspect(2.5); plot2d("p(x)", -20, 20):
! [images/23030630036_Marcelline
function f(x) := 2^x // fungsi eksponensial
f(2), f(0), f(-1), f(-10)
4 1 0.5 0.0009765625
plot2d("f(x)", -50, 50):
! [images/23030630036_Marcelline
function g(x) := 4x+5 // fungsi linear
g(-100), g(-20), g(0), g(20)
-395 -75 5 85
plot2d("g(x)", -50, 50):
! [images/23030630036_Marcelline
f(g(0)), g(f(0))
32 9
plot2d("f(g(x))", -50, 50):

```

!images/23030630036\_Marcelline

Trigonometri

Kata trigonometri berasal dari bahasa Yunani, di mana trigonon berarti “tiga sudut” dan metron berarti “mengukur”.

Membuat grafik trigonometri

Bentuk grafik dari

$$\sin(x)$$

plot2d("sin(x)",-2pi,2pi,grid=1):

!images/23030630036\_Marcelline

Bentuk grafik fungsi

$$\cos(x)$$

plot2d("cos(x)",-2pi,2pi,grid=1);

title("Plot\_Cos(x)");

xlabel("Sumbu X");

ylabel("Sumbu Y");

!images/23030630036\_Marcelline

Fungsi Eksponensial

Definisi

Secara umum, fungsi eksponen didefinisikan sebagai:

$$3^2, \exp(2\ln(3))$$

9 9

Sifat-sifat Fungsi Eksponensial

Apabila  $a > 0$  dan  $b > 0$ ,  $x$  dan  $y$  adalah bilangan rasional, maka

$$a^x b^y = a^{x+y}$$

$$\frac{a^x}{b^y} = a^{x-y}$$

a:=2

2

x:=3

3

y:=4

4

ri:=(a<sup>x</sup>)<sup>y</sup>

4096

Menghitung Limit

Perhitungan limit pada EMT dapat dilakukan dengan menggunakan fungsi Maxima, yakni "limit". Fungsi "limit" dapat digunakan untuk menghitung limit fungsi dalam bentuk ekspresi maupun fungsi yang sudah didefinisikan sebelumnya. Nilai limit dapat dihitung pada sebarang nilai atau pada tak hingga (-inf, minf, dan inf). Limit kiri dan limit kanan juga dapat dihitung, dengan cara

memberi opsi "plus" atau "minus". Hasil limit dapat berupa nilai, "und" (tak definisi), "ind" (tak tentu namun terbatas), "infinity" (kompleks tak hingga).

Perhatikan beberapa contoh berikut. Perhatikan cara menampilkan perhitungan secara lengkap, tidak hanya menampilkan hasilnya saja.

*showev('limit(sqrt(x^2-3x)/(x+1),x,inf))*

$$\lim_{x \rightarrow \infty} \frac{\sqrt{x^2 - 3x}}{x + 1} = 1$$

*limit((x^3-13x^2+51x-63)/(x^3-4x^2-3x+18),x,3)*

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

—

$$\lim_{x \rightarrow 3} \frac{x^3 - 13x^2 + 51x - 63}{x^3 - 4x^2 - 3x + 18} = -\frac{4}{5}$$

Fungsi tersebut diskontinu di titik x=3. Berikut adalah grafik fungsinya.

*aspect(1.5); plot2d("(x^3-13x^2+51x-63)/(x^3-4x^2-3x+18)",0,4); plot2d(3,-4/5,points,style="ow",add):*

*![images/23030630036\_Marcelline  
limit(2xsin(x)/(1-cos(x)),x,0)*

$$4$$

—

$$2 \left( \lim_{x \rightarrow 0} \frac{x \sin x}{1 - \cos x} \right) = 4$$

Fungsi tersebut diskontinu di titik x=0. Berikut adalah grafik fungsinya.

*plot2d("2xsin(x)/(1-cos(x))",-pi,pi); plot2d(0,4,points,style="ow",add):*

*![images/23030630036\_Marcelline  
limit(cot(7h)/cot(5h),h,0)*

$$\frac{5}{7}$$

—

$$\lim_{h \rightarrow 0} \frac{\cot(7h)}{\cot(5h)} = \frac{5}{7}$$

Fungsi tersebut juga diskontinu (karena tidak terdefinisi) di x=0. Berikut adalah grafiknya.

*plot2d("cot(7x)/cot(5x)",-0.001,0.001); plot2d(0,5/7,points,style="ow",add):*

*![images/23030630036\_Marcelline  
showev('limit(((x/8)^(1/3)-1)/(x-8),x,8))*

$$\lim_{x \rightarrow 8} \frac{\frac{x^{\frac{1}{3}}}{2} - 1}{x - 8} = \frac{1}{24}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

plot2d("(x^(1/3))/2 - 1)/(x - 8)", -5, 10); plot2d(8, 1/24, points, style = "ow", add) :

![images/23030630036\_Marcelline  
showev('limit(1/(2x - 1), x, 0))

$$\lim_{x \rightarrow 0} \frac{1}{2x - 1} = -1$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

plot2d("1/(2x-1)", -0.001, 0.001); plot2d(0, -1, points, style="ow", add):

![images/23030630036\_Marcelline  
showev('limit((x^2 - 3x - 10)/(x - 5), x, 5))

$$\lim_{x \rightarrow 5} \frac{x^2 - 3x - 10}{x - 5} = 7$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

plot2d("(x^2-3x-10)/(x-5)", -20, 20); plot2d(5, 7, points, style = "ow", add) :

![images/23030630036\_Marcelline  
showev('limit(sqrt(x^2 + x) - x, x, inf))

$$\lim_{x \rightarrow \infty} \sqrt{x^2 + x} - x = \frac{1}{2}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

plot2d("sqrt(x^2 + x)", -0.3, 2) :

![images/23030630036\_Marcelline  
showev('limit(abs(x - 1)/(x - 1), x, 1, minus))

$$\lim_{x \uparrow 1} \frac{|x - 1|}{x - 1} = -1$$

Hitung limit di atas untuk x menuju 1 dari kanan.

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

showev('limit(abs(x - 1)/(x - 1), x, 1, plus))

$$\lim_{x \downarrow 1} \frac{|x - 1|}{x - 1} = 1$$

showev('limit(sin(x)/x, x, 0))

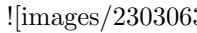
$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

plot2d("sin(x)/x", -pi, pi); plot2d(0, 1, points, style="ow", add):

![images/23030630036\_Marcelline  
showev('limit(sin(x^3)/x, x, 0))

$$\lim_{x \rightarrow 0} \frac{\sin x^3}{x} = 0$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
plot2d("sin(x^3)/x", -pi, pi); plot2d(0, 0, points, style = "ow", add) :

showev('limit(log(x), x, minf))
```

$$\lim_{x \rightarrow -\infty} \log x = \text{infinity}$$

```
showev('limit((-2)^x, x, inf))
```

$$\lim_{x \rightarrow \infty} (-2)^x = \text{infinity}$$

```
showev('limit(t - sqrt(2 - t), t, 2, minus))
```

$$\lim_{t \uparrow 2} t - \sqrt{2 - t} = 2$$

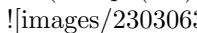
```
showev('limit(t - sqrt(2 - t), t, 2, plus))
```

$$\lim_{t \downarrow 2} t - \sqrt{2 - t} = 2$$

```
showev('limit(t - sqrt(2 - t), t, 5, plus))//Perhatikanhasilnya
```

$$\lim_{t \downarrow 5} t - \sqrt{2 - t} = 5 - \sqrt{3}i$$

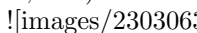
```
plot2d("x-sqrt(2-x)", 0, 2):
```

```
! 
showev('limit((x^2 - 9)/(2x^2 - 5x - 3), x, 3))
```

$$\lim_{x \rightarrow 3} \frac{x^2 - 9}{2x^2 - 5x - 3} = \frac{6}{7}$$

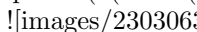
Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
plot2d("(x^2-9)/(2x^2-5x-3)", -0.2, 4.5, -0.2, 4.4); plot2d(3, 6/7, points, style =
"ow", add) :
```

```
! 
showev('limit((1 - cos(x))/x, x, 0))
```

$$\lim_{x \rightarrow 0} \frac{1 - \cos x}{x} = 0$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
plot2d("(1-cos (x))/x", -1, 1); plot2d(0, 0, points, style="ow", add):
! 
showev('limit((x^2 + abs(x))/(x^2 - abs(x)), x, 0))
```

$$\lim_{x \rightarrow 0} \frac{|x| + x^2}{x^2 - |x|} = -1$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
plot2d("(x^2+abs(x))/(x^2-abs(x))", -1.2, 1.2, -100, 1); plot2d(0, -1, points, style =
"ow", add) :
```

```
![images/23030630036_Marcelline
showev('limit((1 + 1/x)^x, x, inf))
```

$$\lim_{x \rightarrow \infty} \left( \frac{1}{x} + 1 \right)^x = e$$

```
aspect(2); plot2d("(1+1/x)^x", 0, 1000) :
```

```
![images/23030630036_Marcelline
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))
```

$$\lim_{x \rightarrow 0} (x + 1)^{\frac{1}{x}} = e$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
plot2d("(1+x)^(1/x)", 0, 10) :
```

```
![images/23030630036_Marcelline
showev('limit((x/(x + k))^x, x, inf))
```

$$\lim_{x \rightarrow \infty} \left( \frac{x}{x + k} \right)^x = e^{-k}$$

```
showev('limit((E^x - E^2)/(x - 2), x, 2))
```

$$\lim_{x \rightarrow 2} \frac{e^x - e^2}{x - 2} = e^2$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
plot2d("(E^x - E^2)/(x - 2)", 60, 110) :
```

```
![images/23030630036_Marcelline
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/23030630036_Marcelline
plot2d("sin(1/x)", -1, 1):
![images/23030630036_Marcelline
```

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.

*showev('limit((x<sup>2</sup> - x - 6)/(x - 3), x, 3))*

$$\lim_{x \rightarrow 3} \frac{x^2 - x - 6}{x - 3} = 5$$

*plot2d("(x<sup>2</sup> - x - 6)/(x - 3)", 2, 5); plot2d(3, 5, points, style = "ow", add) :*

*![images/23030630036\_Marcelline*

*showev('limit(x<sup>2</sup> - (cos(x)/10000), x, 0))*

$$\lim_{x \rightarrow 0} x^2 - \frac{\cos x}{10000} = -\frac{1}{10000}$$

*plot2d("x<sup>2</sup> - (cos(x)/10000)", -0.02, 0.05); plot2d(0, -1/10000, points, style = "ow", add) :*

*![images/23030630036\_Marcelline*

*showev('limit(tan(u)/u, u, pi/2))*

$$\lim_{u \rightarrow \frac{\pi}{2}} \frac{\tan u}{u} = \text{infinity}$$

*showev('limit(x/abs(x), x, 0, minus))*

$$\lim_{x \uparrow 0} \frac{x}{|x|} = -1$$

*showev('limit(x/abs(x), x, 0, plus))*

$$\lim_{x \downarrow 0} \frac{x}{|x|} = 1$$

*plot2d("x/abs(x)", -0.2, 0.2, -1.2, 1.2):*

*![images/23030630036\_Marcelline*

*showev('limit((sin(x))<sup>(cos(x))</sup>, x, pi/2))*

$$\lim_{x \rightarrow \frac{\pi}{2}} (\sin x)^{\cos x} = 1$$

*plot2d("(sin(x))<sup>(cos(x))</sup>", 1, 2.3); plot2d(pi/2, 1, points, style = "ow", add) :*

*![images/23030630036\_Marcelline*

Turunan Fungsi

Definisi turunan:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

Berikut adalah contoh-contoh menentukan turunan fungsi dengan menggunakan definisi turunan (limit).

*showev('limit(((x+h)^2 - x^2)/h, h, 0))//turunanx^2*

$$\lim_{h \rightarrow 0} \frac{(x+h)^2 - x^2}{h} = 2x$$

*p = expand((x+h)^2 - x^2)|simplify;p //pembilang dijabarkan dan disederhanakan*

$$2hx + h^2$$

*q = ratsimp(p/h); q//ekspresiyangakandihitunglimitnyadisederhanakan*

$$2x + h$$

*limit(q, h, 0)//nilailimitsebagaiturunan*

$$2x$$

*showev('limit(((x+h)^n - x^n)/h, h, 0))//turunanx^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.

*showev('limit((sin(x+h) - sin(x))/h, h, 0))//turunansin(x)*

$$\lim_{h \rightarrow 0} \frac{\sin(x+h) - \sin x}{h} = \cos x$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut

benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, ekspansikan  $\sin(x+h)$  dengan menggunakan rumus jumlah dua sudut.

*showev('limit((log(x+h) - log(x))/h, h, 0))//turunanlog(x)*

$$\lim_{h \rightarrow 0} \frac{\log(x+h) - \log x}{h} = \frac{1}{x}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut

benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, gunakan sifat-sifat logaritma dan hasil limit pada bagian sebelumnya di atas.



*showev('limit((1/(x+h) - 1/x)/h, h, 0))//turunan1/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))//turunanf(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))//turunanf(x) = e^x...*

Maxima bermasalah dengan limit:

$$\lim_{h \rightarrow 0} \frac{e^{x+h} - e^x}{h}.$$

Oleh karena itu diperlukan trik khusus agar hasilnya benar.

*showev('limit((E^h - 1)/h, h, 0))*

$$\lim_{h \rightarrow 0} \frac{e^h - 1}{h} = 1$$

*showev('factor(E(x+h) - E^x))*

$$\text{factor}(e^{x+h} - e^x) = (e^h - 1) e^x$$

*showev('limit(factor((E(x+h) - E^x)/h), h, 0))//turunanf(x) = e^x*

$$\left( \lim_{h \rightarrow 0} \frac{e^h - 1}{h} \right) e^x = e^x$$

function f(x) = x<sup>x</sup>

x x

*showev('limit(f(x), x, 0))*

$$\lim_{x \rightarrow 0} x^x = 1$$

Silakan Anda gambar kurva

$$y = x^x.$$

*aspect(3); plot2d("x^x", -0.2, 2); plot2d(0, 1, points, style = "ow", add) :*

!images/23030630036\_Marcelline

*showev('limit((f(x+h) - f(x))/h, h, 0))//turunanf(x) = x^x*

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h} = \text{infinity}$$

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))//turunan f(x) = x^x`

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h} = x^x (\log x + 1)$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

`forget(x 0) // jangan lupa, lupakan asumsi untuk kembali ke semula`

`[x gt; 0]`

`forget(x;0)`

`[x lt; 0]`

`facts()`

`[]`

`showev('limit((asin(x+h)-asin(x))/h,h,0))//turunan arcsin(x)`

$$\lim_{h \rightarrow 0} \frac{\arcsin(x+h) - \arcsin x}{h} = \frac{1}{\sqrt{1-x^2}}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

`showev('limit((tan(x+h)-tan(x))/h,h,0))//turunan tan(x)`

$$\lim_{h \rightarrow 0} \frac{\tan(x+h) - \tan x}{h} = \frac{1}{\cos^2 x}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

`function f(x) = sinh(x) // definisikan f(x)=sinh(x)`

`sinh(x)`

`function df(x) = limit((f(x+h)-f(x))/h,h,0); df(x)//df(x) = f'(x)`

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

Hasilnya adalah  $\cosh(x)$ , karena

$$\frac{e^x + e^{-x}}{2} = \cosh(x).$$

`aspect(2); plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]):`

`![images/23030630036_Marcelline`

`function f(x) = sin(3x^5 + 7)^2`

`2 5 sin (3 x + 7)`

```
diff(f,3), diffc(f,3)
1198.32948904 1198.72863721
Apakah perbedaan diff dan diffc?
- diff : digunakan untuk menghitung turunan numerik (diskrit), terutama
untuk data atau fungsi diskrit
- diffc : digunakan untuk menghitung turunan simbolik (kontinu), terutama
untuk ekspresi atau fungsi yang ditulis secara simbolik
showev('diff(f(x),x))
```

$$\frac{d}{dx} \sin^2(3x^5 + 7) = 30x^4 \cos(3x^5 + 7) \sin(3x^5 + 7)$$

$$\%at \left( \frac{d}{dx} \sin^2(3x^5 + 7), x = 3 \right) = 2430 \cos 736 \sin 736$$

```
float(
```

$$\%at \left( \frac{d^{1.0}}{dx^{1.0}} \sin^2(3.0x^5 + 7.0), x = 3.0 \right) = 1198.728637211748$$

```
plot2d(f,0,3.1):
```

```
![images/23030630036_Marcelline
function f(x)=5cos(2x)-2xsin(2x) // mendefinisikan fungsi f
5 cos(2 x) - 2 x sin(2 x)
function df(x)=diff(f(x),x) // fd(x) = f'(x)
- 12 sin(2 x) - 4 x cos(2 x)
'f(1)=f(1),float(f(1)), 'f(2)=f(2),float(f(2)) // nilai f(1) dan f(2)
```

```
-0.2410081230863468
```

```
![images/23030630036_Marcelline
![images/23030630036_Marcelline
![images/23030630036_Marcelline
xp=solve('df(x)',1,2,0) // solusi f'(x)=0 pada interval [1, 2]
1.35822987384
df(xp), f(xp) // cek bahwa f'(xp)=0 dan nilai ekstrim di titik tersebut
0 -5.67530133759
plot2d(['f(x)', 'df(x)'],0,2pi,color=[blue,red]): //grafik fungsi dan turunan-nya
```

```
![images/23030630036_Marcelline
Perhatikan titik-titik "puncak" grafik y=f(x) dan nilai turunan pada saat grafik fungsinya mencapai titik "puncak" tersebut.
```

Latihan

Bukalah buku Kalkulus. Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan di EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi). Untuk setiap fungsi, tentukan turunannya dengan menggunakan definisi turunan (limit),

menggunakan perintah diff, dan secara manual (langkah demi langkah yang dihitung dengan Maxima) seperti contoh-contoh di atas. Gambar grafik fungsi asli dan fungsi turunannya pada sumbu koordinat yang sama.

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

```
2 x
```

```
function df(x) = limit((f(x+h)-f(x))/h,h,0); df(x)/df(x) = f'(x)
```

$$2x$$

```
showev('limit((f(x+h)-f(x))/h,h,0))/caralainmencariturunan
```

$$\lim_{h \rightarrow 0} \frac{(x+h)^2 - x^2}{h} = 2x$$

```
aspect(2.4); plot2d(["f(x)","df(x)"],-1,3,color=[blue,red]): //grafik fungsi dan turunannya
```

```
![images/23030630036_Marcelline
```

---

```
function f(x) = x^3 - 3x^2 + 7x + 8
```

```
3 2 x - 3 x + 7 x + 8
```

```
showev('limit((f(x+h)-f(x))/h,h,0))/caralainmencariturunan
```

$$\lim_{h \rightarrow 0} \frac{(x+h)^3 - 3(x+h)^2 - x^3 + 3x^2 + 7(x+h) - 7x}{h} = 3x^2 - 6x + 7$$

```
aspect(2.2); plot2d(["f(x)","df(x)"],-1.75,1,color=[blue,red]): //grafik fungsi dan turunannya
```

```
![images/23030630036_Marcelline
```

---

```
function f(x) = sqrt(5x+1)
```

```
sqrt(5 x + 1)
```

```
function df(x) = limit((f(x+h)-f(x))/h,h,0); df(x)/df(x) = f'(x)
```

$$\frac{5}{2\sqrt{5x+1}}$$

```
aspect(2.75); plot2d(["f(x)","df(x)"],-0.75,4,color=[blue,red]): //grafik fungsi dan turunannya
```

```
![images/23030630036_Marcelline
```

---

```
function f(x) = 3sin(2x)
```

```
3 sin(2 x)
```

```
function df(x) = limit((f(x+h)-f(x))/h,h,0); df(x)/df(x) = f'(x)
```

$$6 \cos(2x)$$

```
aspect(3); plot2d(["f(x)","df(x)"],-2pi,2pi,color=[blue,red]): //grafik fungsi dan turunannya
```

![images/23030630036\_Marcelline

---

```
function f(x) = (cos(x^2 + 1))^3
3 2 cos (x + 1)
function df(x) = limit((f(x+h)-f(x))/h,h,0); df(x)/df(x) = f'(x)
```

$$3 \left( -2 \cos 1 x \sin x^2 - 2 \sin 1 x \cos x^2 \right) \cos^2 (x^2 + 1)$$

```
plot2d(["f(x)", "df(x)"],-pi,pi,color=[blue,red]): //grafik fungsi dan turunannya
```

![images/23030630036\_Marcelline

---

```
function f(x) = xsinh(x)
x sinh(x)
function df(x) = limit((f(x+h)-f(x))/h,h,0); df(x)/df(x) = f'(x)
```

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

```
plot2d(["f(x)", "df(x)"],-4.75,2pi,color=[blue,red]): //grafik fungsi dan turunannya
```

![images/23030630036\_Marcelline

Integral

EMT dapat digunakan untuk menghitung integral, baik integral tak tentu maupun integral tentu. Untuk integral tak tentu (simbolik) sudah tentu EMT menggunakan Maxima, sedangkan untuk perhitungan integral tentu EMT sudah menyediakan beberapa fungsi yang mengimplementasikan algoritma kuadratur (perhitungan integral tentu menggunakan metode numerik).

Pada notebook ini akan ditunjukkan perhitungan integral tentu dengan menggunakan Teorema Dasar Kalkulus:

$$\int_a^b f(x) dx = F(b) - F(a), \quad \text{dengan } F'(x) = f(x).$$

Fungsi untuk menentukan integral adalah integrate. Fungsi ini dapat digunakan untuk menentukan, baik integral tentu maupun tak tentu (jika fungsinya memiliki antiderivatif). Untuk perhitungan integral tentu fungsi integrate menggunakan metode numerik (kecuali fungsinya tidak integrabel, kita tidak akan menggunakan metode ini).

```
showev('integrate(x^n,x))
```

Answering "Is n equal to -1?" with "no"

$$\int x^n dx = \frac{x^{n+1}}{n+1}$$

```
showev('integrate(1/(1+x),x))
```

$$\int \frac{1}{x+1} dx = \log(x+1)$$

*showev('integrate(1/(1 + x^2), x))*

$$\int \frac{1}{x^2 + 1} dx = \arctan x$$

*showev('integrate(1/sqrt(1 - x^2), x))*

$$\int \frac{1}{\sqrt{1 - x^2}} dx = \arcsin x$$

*showev('integrate(sin(x), x, 0, pi))*

$$\int_0^\pi \sin x dx = 2$$

*plot2d("sin(x)", 0, 2pi):*

!images/23030630036\_Marcelline

*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)
2 - x E
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/23030630036_Marcelline
Integral tentu
```

$$\int_0^\pi e^{-x^2} dx$$

dapat dihampiri dengan jumlah luas persegi-persegi panjang di bawah kurva  $y=f(x)$  tersebut. Langkah-langkahnya adalah sebagai berikut.

```
t = makelist(a,a,0,pi-0.1,0.1); // t sebagai list untuk menyimpan nilai-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, inf))
```

$$\int_0^{\infty} e^{-x^2} dx = \frac{\sqrt{\pi}}{2}$$

Tunjukkan kebenaran hasil di atas!

Berikut adalah contoh lain fungsi yang tidak memiliki antiderivatif, sehingga integral tentunya hanya dapat dihitung dengan metode numerik.

```
function f(x) = x^x
```

```
x x
```

```
showev('integrate(f(x), x, 0, 1))
```

$$\int_0^1 x^x dx = \int_0^1 x^x dx$$

```
x=0:0.1:1-0.01; plot2d(x,f(x+0.01), bar); plot2d("f(x)",0,1, add):
```

```
![images/23030630036_Marcelline
```

Maxima gagal menghitung integral tentu tersebut secara langsung menggunakan perintah integrate. Berikut kita lakukan seperti contoh sebelumnya untuk mendapat hasil atau pendekatan nilai integral tentu tersebut.

```
t = makelist(a,a,0,1-0.01,0.01);
```

```
fx = makelist(f(t[i]+0.01),i,1,length(t));
```

$$\int_0^1 x^x dx = 0.7834935879025506$$

Apakah hasil tersebut cukup baik? perhatikan gambarnya.

```
function f(x) = sin(3x^5 + 7)^2
```

```
2 5 sin (3 x + 7)
```

```
integrate(f,0,1)
```

```
0.542581176074
```

```
showev('integrate(f(x), x, 0, 1))
```

$$\int_0^1 \sin^2(3x^5 + 7) dx = \frac{\Gamma\left(\frac{1}{5}\right) \sin 14 \sin\left(\frac{\pi}{10}\right)}{10 6^{\frac{1}{5}}} - \frac{\left(\left(6^{\frac{4}{5}} \text{gamma\_incomplete}\left(\frac{1}{5}, 6i\right) + 6^{\frac{4}{5}} \text{gamma\_incomplete}\left(\frac{1}{5}, -\right)\right)}{10 6^{\frac{1}{5}}}\right)}{10 6^{\frac{1}{5}}}$$

```
float(
```

$$\int_{0.0}^{1.0} \sin^2(3.0 x^5 + 7.0) dx = 0.09820784258795788 - 0.008333333333333333 (0.3090169943749474 (0.136737218$$

```
showev('integrate(xexp(-x), x, 0, 1))//Integraltentu(eksak)
```

$$\int_0^1 x e^{-x} dx = 1 - 2 e^{-1}$$



Aplikasi Integral Tentu

```
plot2d("x^3 - x", -0.1, 1.1); plot2d(" - x^2", add); ... b = solve("x^3 - x +
x^2", 0.5); x = linspace(0, b, 200); xi = flip(x); ... plot2d(x|xi, x^3 - x| - xi^2, filled, style =
" |", fillcolor = 1, add) : //Plotdaerahantara2kurva
![images/23030630036_Marcelline
a=solve("x^3 - x + x^2", 0), b = solve("x^3 - x + x^2", 1) //absistitik - titik potong kedua kurva
0 0.61803398875
integrate("(-x^2) - (x^3 - x)", a, b) //luas daerah yang diarsir
0.0758191713542
```

Hasil tersebut akan kita bandingkan dengan perhitungan secara analitik.  
 $a = \text{solve}((-x^2) - (x^3 - x), x); a //$  menentukan absis titik potong kedua kurva  
 secara eksak

$$\left[ x = \frac{-\sqrt{5} - 1}{2}, x = \frac{\sqrt{5} - 1}{2}, x = 0 \right]$$

$\text{showev}('integrate(-x^2 - x^3 + x, x, 0, (\text{sqrt}(5) - 1)/2)) // \text{Nilai integral secara eksak}$

$$\int_0^{\frac{\sqrt{5}-1}{2}} -x^3 - x^2 + x \, dx = \frac{13 - 5^{\frac{3}{2}}}{24}$$

$\text{float}(\text{$

$$\int_{0.0}^{0.6180339887498949} -1.0 x^3 - 1.0 x^2 + x \, dx = 0.07581917135421037$$

Panjang Kurva

Hitunglah panjang kurva berikut ini dan luas daerah di dalam kurva tersebut.

$$\gamma(t) = (r(t) \cos(t), r(t) \sin(t))$$

dengan

$$r(t) = 1 + \sin(3t)/2, \quad 0 \leq t \leq 2\pi.$$

$t = \text{linspace}(0, 2\pi, 1000); r = 1 + \sin(3t)/2; x = r \cos(t); y = r \sin(t); \dots \text{plot2d}(x, y, \text{filled}, \text{fillcolor} = \text{red}, \text{style} = " /", r = 1.$   
 // Kita gambar kurvanya terlebih dahulu

```
![images/23030630036_Marcelline
function r(t) = 1 + sin(3t)/2; 'r(t) = r(t)
```

$$r(t) = \frac{\sin(3t)}{2} + 1$$

$\text{function } fx(t) = r(t) \cos(t); 'fx(t) = fx(t)$

$$fx(t) = \cos t \left( \frac{\sin(3t)}{2} + 1 \right)$$

```
function fy(t) = r(t)sin(t); 'fy(t) = fy(t)
```

$$fy(t) = \sin t \left( \frac{\sin(3t)}{2} + 1 \right)$$

```
function ds(t) = trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); 'ds(t)=ds(t)
```

$$ds(t) = \frac{\sqrt{4 \cos(6t) + 4 \sin(3t) + 9}}{2}$$

```
integrate(ds(x),x,0,2pi)/panjang(keliling)kurva
```

$$\frac{\int_0^{2\pi} \sqrt{4 \cos(6x) + 4 \sin(3x) + 9} dx}{2}$$

Maxima gagal melakukan perhitungan eksak integral tersebut.

Berikut kita hitung integralnya secara numerik dengan perintah EMT.

```
integrate("ds(x)",0,2pi)
```

```
9.0749467823
```

Spiral Logaritmik

$$x = e^{ax} \cos x, \quad y = e^{ax} \sin x.$$

```
a=0.1; plot2d("exp(ax)cos(x)", "exp(ax)sin(x)",r=2,xmin=0,xmax=2pi):
```

```
![images/23030630036_Marcelline
```

```
kill(a) // hapus ekspresi a
```

```
done
```

```
function fx(t) = exp(at)cos(t); 'fx(t) = fx(t)
```

$$fx(t) = e^{at} \cos t$$

```
function fy(t) = exp(at)sin(t); 'fy(t) = fy(t)
```

$$fy(t) = e^{at} \sin t$$

```
function df(t) = trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2))); 'df(t)=df(t)
```

$$df(t) = \sqrt{a^2 + 1} e^{at}$$

```
S=integrate(df(t),t,0,2
```

$$\sqrt{a^2 + 1} \left( \frac{e^{2\pi a}}{a} - \frac{1}{a} \right)$$

```
S(a=0.1) // Panjang kurva untuk a=0.1
```

```
8.78817491636
```

Soal:

Tunjukkan bahwa keliling lingkaran dengan jari-jari r adalah  $K=2\pi r$ .

Berikut adalah contoh menghitung panjang parabola.

```
aspect(3); plot2d("x^2", xmin=-1, xmax=1) :
```

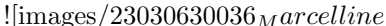
```
![images/23030630036_Marcelline
```

*showev('integrate(sqrt(1 + diff(x^2, x)^2), x, -1, 1))*

$$\int_{-1}^1 \sqrt{4x^2 + 1} \, dx = \frac{\operatorname{asinh} 2 + 2\sqrt{5}}{2}$$

*float(*

$$\int_{-1.0}^{1.0} \sqrt{4.0x^2 + 1.0} \, dx = 2.957885715089195$$

*x=-1:0.2:1; y=x^2; plot2d(x, y); ... plot2d(x, y, points = 1, style = "o", add = 1) :*  


Panjang tersebut dapat dihampiri dengan menggunakan jumlah panjang ruas-ruas garis yang menghubungkan titik-titik pada parabola tersebut.

*i=1:cols(x)-1; sum(sqrt((x[i+1]-x[i])^2 + (y[i+1] - y[i])^2))*  
 2.95191957027

Hasilnya mendekati panjang yang dihitung secara eksak. Untuk mendapatkan hampiran yang cukup akurat, jarak antar titik dapat diperkecil, misalnya 0.1, 0.05, 0.01, dan seterusnya. Cobalah Anda ulangi perhitungannya dengan nilai-nilai tersebut.

Koordinat Kartesius

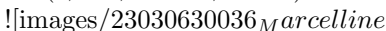
Berikut diberikan contoh perhitungan panjang kurva menggunakan koordinat Kartesius. Kita akan hitung panjang kurva dengan persamaan implisit:

$$x^3 + y^3 - 3xy = 0.$$

$$z = x^3 + y^3 - 3xy; z$$

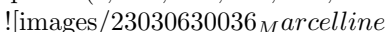
$$y^3 - 3xy + x^3$$

*plot2d(z, r=2, level=0, n=100):*



Kita tertarik pada kurva di kuadran pertama.


*plot2d(z, a=0, b=2, c=0, d=2, level=[-10;0], n=100, contourwidth=3, style="/"):*



Kita selesaikan persamaannya untuk x.

*zwithy = lx, sol = solve(*

$$\left[ x = \frac{3l}{l^3 + 1}, x = 0 \right]$$

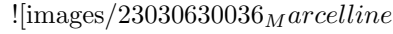


Kita gunakan solusi tersebut untuk mendefinisikan fungsi dengan Maxima.

*function f(l) = rhs(sol[1]); 'f(l) = f(l)*

$$f(l) = \frac{3l}{l^3 + 1}$$

Fungsi tersebut juga dapat digunakan untuk menggambar kurvanya. Ingat, bahwa fungsi tersebut adalah nilai x dan nilai y=l\*x, yakni x=f(l) dan y=l\*f(l).

plot2d(f(x),xf(x),xmin=-0.5,xmax=2,a=0,b=2,c=0,d=2,r=1.5):  

  
Elemen panjang kurva adalah:

$$ds = \sqrt{f'(l)^2 + (lf'(l) + f(l))^2}.$$

function ds(l) = ratsimp(sqrt(diff(f(l),l)^2 + diff(f(l),l)^2)); ds(l)=ds(l)

$$ds(l) = \frac{\sqrt{9l^8 + 36l^6 - 36l^5 - 36l^3 + 36l^2 + 9}}{\sqrt{l^{12} + 4l^9 + 6l^6 + 4l^3 + 1}}$$

integrate(ds(l),l,0,1)

$$\int_0^1 \frac{\sqrt{9l^8 + 36l^6 - 36l^5 - 36l^3 + 36l^2 + 9}}{\sqrt{l^{12} + 4l^9 + 6l^6 + 4l^3 + 1}} dl$$

Integral tersebut tidak dapat dihitung secara eksak menggunakan Maxima. Kita hitung integral tersebut secara numerik dengan Euler. Karena kurva simetris, kita hitung untuk nilai variabel integrasi dari 0 sampai 1, kemudian hasilnya dikalikan 2.

```
2integrate("ds(x)",0,1)
4.91748872168
```

2romberg(ds(x),0,1)// perintah Euler lain untuk menghitung nilai hampiran integral yang sama

```
4.91748872168
```

Perhitungan di atas dapat dilakukan untuk sebarang fungsi x dan y dengan mendefinisikan fungsi EMT, misalnya kita beri nama panjangkurva. Fungsi ini selalu memanggil Maxima untuk menurunkan fungsi yang diberikan.

```
function panjangkurva(fx,fy,a,b) ...
ds=mxm("sqrt(diff(@fx,x)^2+diff(@fy,x)^2)");returnromberg(ds,a,b);endfunction <
/pre > panjangkurva("x","x^2",-1,1)//cekuntukmenghitungpanjangkurvaparabolasebelumnya
2.95788571509
```

Bandingkan dengan nilai eksak di atas.

```
2panjangkurva(mxm("f(x)",mxm("xf(x)"),0,1) // cek contoh terakhir, band-
ingkan hasilnya!
```

```
4.91748872168
```

Kita hitung panjang spiral Archimedes berikut ini dengan fungsi tersebut.

```
plot2d("xcos(x)","xsin(x)",xmin=0,xmax=2pi,square=1):
```

```
!images/23030630036_Marcelline
```

```
panjangkurva("xcos(x)","xsin(x)",0,2pi)
```

```
21.2562941482
```

Berikut kita definisikan fungsi yang sama namun dengan Maxima, untuk perhitungan eksak.

```
kill(ds,x,fx,fy)
```

```
done
```

```
function ds(fx,fy) = sqrt(diff(fx,x)^2 + diff(fy,x)^2)
```

```
2 2 sqrt(diff (fy, x) + diff (fx, x))
```

```
sol = ds(xcos(x),xsin(x)); sol//Kitagunakanuntukmenghitungpanjangkurvaterakhir diatas

$$\sqrt{(\cos x - x \sin x)^2 + (\sin x + x \cos x)^2}$$

sol|trigreduce|expand,integrate(

$$\frac{\operatorname{asinh}(2\pi) + 2\pi\sqrt{4\pi^2 + 1}}{2}$$

```

```
![images/23030630036_Marcelline
21.2562941482
```

Hasilnya sama dengan perhitungan menggunakan fungsi EMT.

Berikut adalah contoh lain penggunaan fungsi Maxima tersebut.

```
plot2d("3x^2-1", "3x^3-1", xmin = -1/sqrt(3), xmax = 1/sqrt(3), square =
1) :
```

```
![images/23030630036_Marcelline
```

```
sol = radcan(ds(3x^2 - 1, 3x^3 - 1));sol
```

$$3x\sqrt{9x^2 + 4}$$

```
showev('integrate(sol, x, 0, 1/sqrt(3))),2float(
```

$$6.0 \int_{0.0}^{0.5773502691896258} x \sqrt{9.0x^2 + 4.0} dx = 2.337835372767141$$

```
![images/23030630036_Marcelline
```

```
reset;
```

Sikloid

Berikut kita akan menghitung panjang kurva lintasan (sikloid) suatu titik pada lingkaran yang berputar ke kanan pada permukaan datar. Misalkan jari-jari lingkaran tersebut adalah r. Posisi titik pusat lingkaran pada saat t adalah:

$$(rt, r).$$

Misalkan posisi titik pada lingkaran tersebut mula-mula (0,0) dan posisinya pada saat t adalah:

$$(r(t - \sin(t)), r(1 - \cos(t))).$$

Berikut kita plot lintasan tersebut dan beberapa posisi lingkaran ketika t=0, t=pi/2, t=r\*pi.

$$x = r(t - \sin(t))$$

$$r(t - \sin(t))$$

$$y = r(1 - \cos(t))$$

$$r(1 - \cos(t))$$

Berikut kita gambar sikloid untuk r=1.

```
kill(ds,x,fx,fy)
```

```
done
```

```

ex = x-sin(x); ey = 1-cos(x); aspect(1);
plot2d(ex,ey,xmin=0,xmax=4pi,square=1); ... plot2d("2+cos(x)","1+sin(x)",
xmin=0, xmax=2pi, add, color=blue); ... plot2d([2,ex(2)],[1,ey(2)], color=red,
add); ... plot2d(ex(2), ey(2), points, add, color=red); ... plot2d("2pi+cos(x)","1+sin(x)",
xmin=0, xmax=2pi, add, color=blue); ... plot2d([2pi,ex(2pi)],[1,ey(2pi)],
color=red, add); ... plot2d(ex(2), ey(2), points, add, color=red):
!images/23030630036_Marcelline

```

Berikut dihitung panjang lintasan untuk 1 putaran penuh. (Jangan salah menduga bahwa panjang lintasan 1 putaran penuh sama dengan keliling lingkaran!)

```

ds = radcan(sqrt(diff(ex,x)^2+diff(ey,x)^2));ds=trigsimp(ds) // elemen pan-
jang kurva sikloid

```

$$\sqrt{\sin^2 x + \cos^2 x - 2 \cos x + 1} = \sqrt{2 - 2 \cos x}$$

```

ds = trigsimp(ds); ds

```

$$\sqrt{2 - 2 \cos x}$$

```

showev('integrate(ds,x,0,2pi)')//hitungpanjangsikloidsatuputaranpenuh

```

$$\int_0^{2\pi} \sqrt{2 - 2 \cos x} \, dx = 8$$

```

integrate(mxm("ds"),0,2pi) // hitung secara numerik
8

```

```

romberg(mxm("ds"),0,2pi) // cara lain hitung secara numerik
8

```

Perhatikan, seperti terlihat pada gambar, panjang sikloid lebih besar daripada keliling lingkarannya, yakni:

$$2\pi.$$

Kurvatur (Kelengkungan) Kurva

image: Osculating.png

Aslinya, kelengkungan kurva diferensiabel (yakni, kurva mulus yang tidak lancip) di titik P didefinisikan melalui lingkaran oskulasi (yaitu, lingkaran yang melalui titik P dan terbaik memperkirakan, paling banyak menyinggung kurva di sekitar P). Pusat dan radius kelengkungan kurva di P adalah pusat dan radius lingkaran oskulasi. Kelengkungan adalah kebalikan dari radius kelengkungan:

$$\kappa = \frac{1}{R}$$

dengan R adalah radius kelengkungan. (Setiap lingkaran memiliki kelengkungan ini pada setiap titiknya, dapat diartikan, setiap lingkaran berputar  $2\pi$  sejauh  $2\pi R$ .)

Definisi ini sulit dimanipulasi dan dinyatakan ke dalam rumus untuk kurva umum. Oleh karena itu digunakan definisi lain yang ekuivalen.

Definisi Kurvatur dengan Fungsi Parametrik Panjang Kurva

Setiap kurva diferensiabel dapat dinyatakan dengan persamaan parametrik terhadap panjang kurva s:

$$\gamma(s) = (x(s), y(s)),$$

dengan x dan y adalah fungsi riil yang diferensiabel, yang memenuhi:

$$\|\gamma'(s)\| = \sqrt{x'(s)^2 + y'(s)^2} = 1.$$

Ini berarti bahwa vektor singgung

$$\mathbf{T}(s) = (x'(s), y'(s))$$

memiliki norm 1 dan merupakan vektor singgung satuan.

Apabila kurvanya memiliki turunan kedua, artinya turunan kedua x dan y ada, maka  $\mathbf{T}'(s)$  ada. Vektor ini merupakan normal kurva yang arahnya menuju pusat kurvatur, norm-nya merupakan nilai kurvatur (kelengkungan):

$$\mathbf{T}(s) = \gamma'(s), \mathbf{T}^2(s) = 1 \text{ (konstanta)} \Rightarrow \mathbf{T}'(s) \cdot \mathbf{T}(s) = 0 \kappa(s) = \|\mathbf{T}'(s)\| = \|\gamma''(s)\| = \sqrt{x''(s)^2 + y''(s)^2}.$$

Nilai

$$R(s) = \frac{1}{\kappa(s)}$$

disebut jari-jari (radius) kelengkungan kurva.

Bilangan riil

$$k(s) = \pm \kappa(s)$$

disebut nilai kelengkungan bertanda.

Contoh:

Akan ditentukan kurvatur lingkaran

$$x = r \cos t, \quad y = r \sin t.$$

`fx = rcos(t); fy =rsin(t);`

`assume(t 0,r 0); s=integrate(sqrt(diff(fx,t)^2+diff(fy,t)^2),t,0,t); s//elemenpanjangkurva,panjangbusur`

`r t`

`kill(s); fx = rcos(s/r); fy =rsin(s/r); // definisi ulang persamaan parametrik terhadap s dengan substitusi t=s/r`

`k = trigsimp(sqrt(diff(fx,s,2)^2+diff(fy,s,2)^2));k // nilai kurvatur lingkaran dengan menggunakan definisi di atas`

$$\frac{1}{r}$$

Untuk representasi parametrik umum, misalkan

$$x = x(t), \quad y = y(t)$$

merupakan persamaan parametrik untuk kurva bidang yang terdiferensialkan dua kali. Kurvatur untuk kurva tersebut didefinisikan sebagai

$$\kappa = \frac{d\phi}{ds} = \frac{\frac{d\phi}{dt}}{\frac{ds}{dt}} \quad (\phi \text{ adalah sudut kemiringan garis singgung dan } s \text{ adalah panjang kurva}) = \frac{\frac{d\phi}{dt}}{\sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2}} = \frac{1}{\sqrt{\left(\frac{dx}{d\phi}\right)^2 + \left(\frac{dy}{d\phi}\right)^2}}$$

Selanjutnya, pembilang pada persamaan di atas dapat dicari sebagai berikut.

$$\sec^2 \phi \frac{d\phi}{dt} = \frac{d}{dt} (\tan \phi) = \frac{d}{dt} \left( \frac{dy}{dx} \right) = \frac{d}{dt} \left( \frac{dy/dt}{dx/dt} \right) = \frac{d}{dt} \left( \frac{y'(t)}{x'(t)} \right) = \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2} \cdot \frac{d\phi}{dt} = \frac{1}{\sec^2 \phi} \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2}$$

Jadi, rumus kurvatur untuk kurva parametrik

$$x = x(t), \quad y = y(t)$$

adalah

$$\kappa(t) = \frac{x'(t)y''(t) - x''(t)y'(t)}{(x'(t)^2 + y'(t)^2)^{3/2}}.$$

Jika kurvanya dinyatakan dengan persamaan parametrik pada koordinat kutub

$$x = r(\theta) \cos \theta, \quad y = r(\theta) \sin \theta,$$

maka rumus kurvturnya adalah

$$\kappa(\theta) = \frac{r(\theta)^2 + 2r'(\theta)^2 - r(\theta)r''(\theta)}{(r'(\theta)^2 + r(\theta)^2)^{3/2}}.$$

(Silakan Anda turunkan rumus tersebut!)

Contoh:

Lingkaran dengan pusat (0,0) dan jari-jari r dapat dinyatakan dengan persamaan parametrik

$$x = r \cos t, \quad y = r \sin t.$$

Nilai kelengkungan lingkaran tersebut adalah

$$\kappa(t) = \frac{x'(t)y''(t) - x''(t)y'(t)}{(x'(t)^2 + y'(t)^2)^{3/2}} = \frac{r^2}{r^3} = \frac{1}{r}.$$

Hasil cocok dengan definisi kurvatur suatu kelengkungan.

Kurva

$$y = f(x)$$

dapat dinyatakan ke dalam persamaan parametrik



$$x = t, y = f(t), \text{ dengan } x'(t) = 1, x''(t) = 0,$$

sehingga kurvturnya adalah

$$\kappa(t) = \frac{y''(t)}{(1 + y'(t)^2)^{3/2}}.$$

Contoh:

Akan ditentukan kurvatur parabola

$$y = ax^2 + bx + c.$$

kill(x,y)

done

function f(x) = ax<sup>2</sup> + bx + c; y=f(x)

$$y = ax^2 + bx + c$$

function k(x) = (diff(f(x),x,2))/(1+diff(f(x),x,2)<sup>3/2</sup>); k(x)=k(x) // kelengkungan parabola

$$k(x) = \frac{2a}{\left((2ax+b)^2 + 1\right)^{\frac{3}{2}}}$$

function f(x) = x<sup>2</sup> + x + 1; y=f(x) // akan kita plot kelengkungan parabola untuk a=b=c=1

$$y = x^2 + x + 1$$

function k(x) = (diff(f(x),x,2))/(1+diff(f(x),x,2)<sup>3/2</sup>); k(x)=k(x) // kelengkungan parabola

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

Berikut kita gambar parabola tersebut beserta kurva kelengkungan, kurva jari-jari kelengkungan dan salah satu lingkaran oskulasi di titik puncak parabola. Perhatikan, puncak parabola dan jari-jari lingkaran oskulasi di puncak parabola adalah

$$(-1/2, 3/4), 1/k(2) = 1/2,$$

sehingga pusat lingkaran oskulasi adalah (-1/2, 5/4).

```
plot2d(["f(x)", "k(x)"], -2, 1, color=[blue, red]); plot2d("1/k(x)", -1.5, 1, color=green, add);
... plot2d("-1/2+1/k(-1/2)cos(x)", "5/4+1/k(-1/2)sin(x)", xmin=0, xmax=2pi, add, color=blue);
![images/23030630036_Marcelline
```

Untuk kurva yang dinyatakan dengan fungsi implisit

$$F(x, y) = 0$$

dengan turunan-turunan parsial

$$F_x = \frac{\partial F}{\partial x}, \quad F_y = \frac{\partial F}{\partial y}, \quad F_{xy} = \frac{\partial}{\partial y} \left( \frac{\partial F}{\partial x} \right), \quad F_{xx} = \frac{\partial}{\partial x} \left( \frac{\partial F}{\partial x} \right), \quad F_{yy} = \frac{\partial}{\partial y} \left( \frac{\partial F}{\partial y} \right),$$

berlaku

$$F_x dx + F_y dy = 0 \text{ atau } \frac{dy}{dx} = -\frac{F_x}{F_y},$$

sehingga kurvturnya adalah

$$\kappa = \frac{F_y^2 F_{xx} - 2F_x F_y F_{xy} + F_x^2 F_{yy}}{(F_x^2 + F_y^2)^{3/2}}.$$

(Silakan Anda turunkan sendiri!)

Contoh 1:

Parabola

$$y = ax^2 + bx + c$$

dapat dinyatakan ke dalam persamaan implisit

$$ax^2 + bx + c - y = 0.$$

function F(x,y) = ax<sup>2</sup> + bx + c - y; F(x,y)

$$-y + ax^2 + bx + c$$

Fx = diff(F(x,y),x), Fxx = diff(F(x,y),x,2), Fy = diff(F(x,y),y), Fxy = diff(diff(F(x,y),x),y),  
Fyy = diff(F(x,y),y,2)

$$2ax + b$$

$$2a$$

$$-1$$

$$0$$

$$0$$

function k(x) = (Fy<sup>2</sup>Fxx - 2FxFyFxy + Fx<sup>2</sup>Fyy)/(Fx<sup>2</sup> + Fy<sup>2</sup>)<sup>(3/2)</sup>; k(x) = k(x)

// kurvatur parabola tersebut

$$k(x) = \frac{2a}{\left((2ax + b)^2 + 1\right)^{\frac{3}{2}}}$$

Hasilnya sama dengan sebelumnya yang menggunakan persamaan parabola biasa.

Latihan

\* Bukalah buku Kalkulus.

\* Cari dan pilih beberapa (paling sedikit 5 fungsi berbeda \* tipe/bentuk/jenis) fungsi dari buku tersebut, kemudian definisikan di \* EMT pada baris-baris perintah berikut (jika perlu tambahkan lagi).

\* Untuk setiap fungsi, tentukan anti turunannya (jika ada), hitunglah \* integral tentu dengan batas-batas yang menarik (Anda tentukan \* sendiri), seperti contoh-contoh tersebut.

\* Lakukan hal yang sama untuk fungsi-fungsi yang tidak dapat \* 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.)

- Apabila fungsi dinyatakan dalam koordinat kutub  $x=f(r,t)$ ,  $y=g(r,t)$ ,  $r=h(t)$ ,  $x=a$  bersesuaian dengan  $t=t_0$  dan  $x=b$  bersesuaian dengan  $t=t_1$ , maka rumus di atas akan menjadi:

$$S = \int_{t_0}^{t_1} \sqrt{x'(t)^2 + y'(t)^2} dt.$$

\* Pilih beberapa kurva menarik (selain lingkaran dan parabola) dari \* buku kalkulus. Nyatakan setiap kurva tersebut dalam bentuk: \* a. koordinat Kartesius (persamaan  $y=f(x)$ ) \* b. koordinat kutub ( $r=r(\theta)$ ) \* c. persamaan parametrik  $x=x(t)$ ,  $y=y(t)$  \* d. persamaan implisit  $F(x,y)=0$

\* Tentukan kurvatur masing-masing kurva dengan menggunakan keempat \* representasi tersebut (hasilnya harus sama).

\* Gambarlah kurva asli, kurva kurvatur, kurva jari-jari lingkaran \* oskulasi, dan salah satu lingkaran oskulasinya.

function  $f(x) = x^3 - 2x^2 + x + 1; f(x)$

$$x^3 - 2x^2 + x + 1$$

*showev('integrate(f(x),x))*

$$\int x^3 - 2x^2 + x + 1 \, dx = \frac{x^4}{4} - \frac{2x^3}{3} + \frac{x^2}{2} + x$$

*showev('integrate(f(x),x,0,4))*

$$\int_0^4 x^3 - 2x^2 + x + 1 \, dx = \frac{100}{3}$$

x=0.01:0.3:2; aspect(2); plot2d(x,f(x+0.2), bar); plot2d("f(x)", -1,2, add):

!images/23030630036\_Marcelline

---

function f(x) = (sin(x+2))^2;f(x)

$$\sin^2(x+2)$$

*showev('integrate(f(x),x))*

$$\int \sin^2(x+2) \, dx = \frac{-\frac{\sin(2(x+2))}{2} + x + 2}{2}$$

*showev('integrate(f(x),x,0,2))*

$$\int_0^2 \sin^2(x+2) \, dx = \frac{\sin 4 - 4}{4} - \frac{\sin 8 - 8}{4}$$

x=-0.02:0.2:2; aspect(2); plot2d(x,f(x+0.003), bar); plot2d("f(x)",-1,4, add):

!images/23030630036\_Marcelline

---

function f(x) = x(sqrt(x+2))^3;f(x)

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

*showev('integrate(f(x),x))*

$$\int x(x+2)^{\frac{3}{2}} \, dx = \frac{2(x+2)^{\frac{7}{2}}}{7} - \frac{4(x+2)^{\frac{5}{2}}}{5}$$

*showev('integrate(f(x),x,0,2))*

$$\int_0^2 x(x+2)^{\frac{3}{2}} \, dx = \frac{2^{\frac{11}{2}}}{35} + \frac{384}{35}$$

x=-0.02:0.2:2; aspect(2); plot2d(x,f(x+0.003), bar); plot2d("f(x)",0,4, add):

!images/23030630036\_Marcelline

---

t = makelist(a,a,0,2-4.01,0.01);  
fx = makelist(f(t[i]+0.01),i,1,length(t));  
function f(x) = x^x + 50;f(x)

$$x^x + 50$$

*showev('integrate(f(x), x))*

$$\int x^x + 50 \, dx = \int e^{x \log x} \, dx + 50x$$

*x=0:0.1:pi-0.01; aspect(3); plot2d(x,f(x+0.01), bar); plot2d("f(x)",0,2pi, add):*

*![images/23030630036\_Marcelline*

*0.01sum(f(x+0.01))*

*17.9656691965*

—

*t = makelist(a,a,0,1-0.01,0.01);*

*fx = makelist(f(t[i]+0.01),i,1,length(t));*

*function f(x) = sin(x)/x; f(x)*

$$\frac{\sin x}{x}$$

*x=-pi:0.07:pi-0.01; plot2d(x,f(x+0.01), bar); plot2d("f(x)",-pi,pi, add):*

*![images/23030630036\_Marcelline*

*0.01sum(f(x+0.01))*

*0.529127679919*

Barisan dan Deret

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>list</sub>" atau "makelist" untuk \* menghasilkan barisan simbolik;

- \* menggunakan fungsi biasa yang inputnya vektor atau barisan;

- \* menggunakan fungsi rekursif.

EMT menyediakan beberapa perintah (fungsi) terkait barisan, yakni:

- \* sum: menghitung jumlah semua elemen suatu barisan

- \* cumsum: jumlah kumulatif suatu barisan

- \* differences: selisih antar elemen-elemen berturutan

EMT juga dapat digunakan untuk menghitung jumlah deret berhingga maupun deret tak hingga, dengan menggunakan perintah (fungsi) "sum". Perhitungan dapat dilakukan secara numerik maupun simbolik dan eksak.

Berikut adalah beberapa contoh perhitungan barisan dan deret menggunakan EMT.

1:10 // barisan sederhana

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

1:2:30

[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29]

Iterasi dan Barisan

EMT menyediakan fungsi `iterate("g(x)", x0, n)` untuk melakukan iterasi

$$x_{k+1} = g(x_k), \quad x_0 = x_0, k = 1, 2, 3, \dots, n.$$

Berikut ini disajikan contoh-contoh penggunaan iterasi dan rekursi dengan EMT. Contoh pertama menunjukkan pertumbuhan dari nilai awal 1000 dengan laju pertambahan 5

```
q=1.05; iterate("xq",1000,n=10)'
1000 1050 1102.5 1157.63 1215.51 1276.28 1340.1 1407.1 1477.46 1551.33
1628.89
```

Contoh berikutnya memperlihatkan bahaya menabung di bank pada masa sekarang! Dengan bunga tabungan sebesar 610000 per bulan, tabungan sebesar 1 juta tanpa diambil selama sekitar 10 tahunan akan habis diambil oleh bank!

```
r=0.005; plot2d(iterate("(1+0.8r)x-10000",1000000,n=130)):
```

![images/23030630036\_Marcelline]

Silakan Anda coba-coba, dengan tabungan minimal berapa agar tidak akan habis diambil oleh bank dengan ketentuan bunga dan biaya administrasi seperti di atas.

Berikut adalah perhitungan minimal tabungan agar aman di bank dengan bunga sebesar  $r$  dan biaya administrasi  $a$ , pajak bunga 20

```
solve(0.8rA - a, A),
```

$$[A = 2500.0]$$

![images/23030630036\_Marcelline]

Berikut didefinisikan fungsi untuk menghitung saldo tabungan, kemudian dilakukan iterasi.

```
function saldo(x,r,a) := round((1+0.8r)x-a,2);
```

```
iterate("saldo",0.005,10,1000,n=6)
```

```
[1000, 994, 987.98, 981.93, 975.86, 969.76, 963.64]
```

```
iterate("saldo",0.005,10,2000,n=6)
```

```
[2000, 1998, 1995.99, 1993.97, 1991.95, 1989.92, 1987.88]
```

```
iterate("saldo",0.005,10,2500,n=6)
```

```
[2500, 2500, 2500, 2500, 2500, 2500, 2500]
```

Tabungan senilai 2,5 juta akan aman dan tidak akan berubah nilai (jika tidak ada penarikan), sedangkan jika tabungan awal kurang dari 2,5 juta, lama kelamaan akan berkurang meskipun tidak pernah dilakukan penarikan uang tabungan.

```
iterate("saldo",0.005,10,3000,n=6)
```

```
[3000, 3002, 3004.01, 3006.03, 3008.05, 3010.08, 3012.12]
```

Tabungan yang lebih dari 2,5 juta baru akan bertambah jika tidak ada penarikan.

Untuk barisan yang lebih kompleks dapat digunakan fungsi `"sequence()"`. Fungsi ini menghitung nilai-nilai  $x[n]$  dari semua nilai sebelumnya,  $x[1], \dots, x[n-1]$  yang diketahui.

Berikut adalah contoh barisan Fibonacci.

```
sequence("x[n-1]+x[n-2]",[1,1],15)
```

[1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610]

Barisan Fibonacci memiliki banyak sifat menarik, salah satunya adalah akar pangkat ke-n suku ke-n akan konvergen ke pecahan emas:

$$\sqrt[5]{(1 + \sqrt{5})}/2 = \text{float}((1 + \sqrt{5})/2)$$

$$\frac{\sqrt{5} + 1}{2} = 1.618033988749895$$

plot2d(sequence("x[n-1]+x[n-2]",[1,1],250)^(1/(1 : 250))) :

![images/23030630036\_Marcelline

Barisan yang sama juga dapat dihasilkan dengan menggunakan loop.

x=ones(500); for k=3 to 500; x[k]=x[k-1]+x[k-2]; end;

Rekursi dapat dilakukan dengan menggunakan rumus yang tergantung pada semua elemen sebelumnya. Pada contoh berikut, elemen ke-n merupakan jumlah (n-1) elemen sebelumnya, dimulai dengan 1 (elemen ke-1). Jelas, nilai elemen ke-n adalah  $2^{(n-2)}$ , untuk  $n = 2, 4, 5, \dots$

sequence("sum(x)",1,10)

[1, 1, 2, 4, 8, 16, 32, 64, 128, 256]

Selain menggunakan ekspresi dalam x dan n, kita juga dapat menggunakan fungsi.

Pada contoh berikut, digunakan iterasi

$$x_n = A \cdot x_{n-1},$$

dengan A suatu matriks 2x2, dan setiap x[n] merupakan matriks/vektor 2x1.

A=[1,1;1,2]; function suku(x,n) := A.x[n-1]

sequence("suku",[1;1],6)

Real 2 x 6 matrix

1 2 5 13 ... 1 3 8 21 ...

Hasil yang sama juga dapat diperoleh dengan menggunakan fungsi perpangkatan matriks "matrixpower()". Cara ini lebih cepat, karena hanya menggunakan perkalian matriks sebanyak  $\log_2(n)$ .

$$x_n = A.x_{n-1} = A^2.x_{n-2} = A^3.x_{n-3} = \dots = A^{n-1}.x_1.$$

sequence("matrixpower(A,n).[1;1]",1,6)

Real 2 x 6 matrix

1 5 13 34 ... 1 8 21 55 ...

Spiral Theodorus

image: Spiral<sub>of</sub>Theodorus.png

Spiral Theodorus (spiral segitiga siku-siku) dapat digambar secara rekursif.

Rumus rekursifnya adalah:

$$x_n = \left(1 + \frac{i}{\sqrt{n-1}}\right) x_{n-1}, \quad x_1 = 1,$$

yang menghasilkan barisan bilangan kompleks.

function g(n) := 1+I/sqrt(n)

Rekursinya dapat dijalankan sebanyak 17 untuk menghasilkan barisan 17 bilangan kompleks, kemudian digambar bilangan-bilangan kompleksnya.

```
reset;
x=sequence("g(n-1)x[n-1]",1,17); aspect(1.75); plot2d(x,r=3.5); textbox(latex("Spiral
Theodorus"),0.4):
!images/23030630036_Marcelline
Selanjutnya dihubungkan titik 0 dengan titik-titik kompleks tersebut meng-
gunakan loop.
for i=1:cols(x); plot2d([0,x[i]], add); end:
!images/23030630036_Marcelline
```

Spiral tersebut juga dapat didefinisikan menggunakan fungsi rekursif, yang tidak memerlukan indeks dan bilangan kompleks. Dalam hal ini digunakan vektor kolom pada bidang.

```
function gstep (v) ...
w=[-v[2];v[1]]; return v+w/norm(w); endfunction ;/pre;
Jika dilakukan it-
erasi 16 kali dimulai dari [1;0] akan didapatkan matriks yang memuat vektor-
vektor dari setiap iterasi.
```

```
x=iterate("gstep",[1;0],16); plot2d(x[1],x[2],r=3.5, points):
!images/23030630036_Marcelline
Kekonvergenan
```

Terkadang kita ingin melakukan iterasi sampai konvergen. Apabila iterasinya tidak konvergen setelah ditunggu lama, Anda dapat menghentikannya dengan menekan tombol [ESC].

```
iterate("cos(x)",1) // iterasi x(n+1)=cos(x(n)), dengan x(0)=1.
0.739085133216
```

Iterasi tersebut konvergen ke penyelesaian persamaan

$$x = \cos(x).$$

Iterasi ini juga dapat dilakukan pada interval, hasilnya adalah barisan interval yang memuat akar tersebut.

```
hasil := iterate("cos(x)", 1,2 ) //iterasi x(n+1)=cos(x(n)), dengan interval
awal (1, 2)
```

```
0.739085133211,0.7390851332133
```

Jika interval hasil tersebut sedikit diperlebar, akan terlihat bahwa interval tersebut memuat akar persamaan  $x=\cos(x)$ .

```
h=expand(hasil,100), cos(h) ;h
```

```
0.73908513309,0.73908513333 1
```

Iterasi juga dapat digunakan pada fungsi yang didefinisikan.

```
function f(x) := (x+2/x)/2
```

Iterasi  $x(n+1)=f(x(n))$  akan konvergen ke akar kuadrat 2.

```
iterate("f",2), sqrt(2)
```

```
1.41421356237 1.41421356237
```

Jika pada perintah `iterate` diberikan tambahan parameter  $n$ , maka hasil iterasinya akan ditampilkan mulai dari iterasi pertama sampai ke- $n$ .



```
iterate("f",2,5)
[2, 1.5, 1.41667, 1.41422, 1.41421, 1.41421]
```

Untuk iterasi ini tidak dapat dilakukan terhadap interval.

```
niterate("f", 1,2 ,5)
[ 1,2 , 1,2 , 1,2 , 1,2 , 1,2 , 1,2 ]
```

Perhatikan, hasil iterasinya sama dengan interval awal. Alasannya adalah perhitungan dengan interval bersifat terlalu longgar. Untuk meningkatkan perhitungan pada ekspresi dapat digunakan pembagian intervalnya, menggunakan fungsi `ieval()`.

```
function s(x) := ieval("(x+2/x)/2",x,10)
```

Selanjutnya dapat dilakukan iterasi hingga diperoleh hasil optimal, dan intervalnya tidak semakin mengecil. Hasilnya berupa interval yang memuat akar persamaan:

$$x = \frac{1}{2} \left( x + \frac{2}{x} \right).$$

Satu-satunya solusi adalah

$$x = \sqrt{2}.$$

```
iterate("s(x)", 1,2 )
1.41421356236,1.41421356239
```

Fungsi `"iterate()"` juga dapat bekerja pada vektor. Berikut adalah contoh fungsi vektor, yang menghasilkan rata-rata aritmetika dan rata-rata geometri.

$$(a_{n+1}, b_{n+1}) = \left( \frac{a_n + b_n}{2}, \sqrt{a_n b_n} \right)$$

Iterasi ke-n disimpan pada vektor kolom `x[n]`.

```
function g(x) := [(x[1]+x[2])/2;sqrt(x[1]x[2])]
```

Iterasi dengan menggunakan fungsi tersebut akan konvergen ke rata-rata aritmetika dan geometri dari nilai-nilai awal.

```
iterate("g",[1;5])
2.60401 2.60401
```

Hasil tersebut konvergen agak cepat, seperti kita cek sebagai berikut.

```
iterate("g",[1;5],4)
1 3 2.61803 2.60403 2.60401 5 2.23607 2.59002 2.60399 2.60401
```

Iterasi pada interval dapat dilakukan dan stabil, namun tidak menunjukkan bahwa limitnya pada batas-batas yang dihitung.

```
iterate("g",[ 1 ; 5 ],4)
Interval 2 x 5 matrix
0.99999999999999778,1.00000000000000022 ... 4.9999999999999911,5.00000000000000089 ...
```

Iterasi berikut konvergen sangat lambat.

$$x_{n+1} = \sqrt{x_n}.$$

```
iterate("sqrt(x)",2,10)
```

[2, 1.41421, 1.18921, 1.09051, 1.04427, 1.0219, 1.01089, 1.00543, 1.00271, 1.00135, 1.00068]

Kekonvergenan iterasi tersebut dapat dipercepat dengan percepatan Steffenson:

```
steffenson("sqrt(x)",2,10)
[1.04888, 1.00028, 1, 1]
```

Iterasi menggunakan Loop yang ditulis Langsung

Berikut adalah beberapa contoh penggunaan loop untuk melakukan iterasi yang ditulis langsung pada baris perintah.

```
x=2; repeat x=(x+2/x)/2; until x^2 = 2; end; x,
1.41421356237
```

Penggabungan matriks menggunakan tanda "—" dapat digunakan untuk menyimpan semua hasil iterasi.

```
v=[1]; for i=2 to 8; v=v—(v[i-1]i); end; v,
[1, 2, 6, 24, 120, 720, 5040, 40320]
```

hasil iterasi juga dapat disimpan pada vektor yang sudah ada.

```
v=ones(1,100); for i=2 to cols(v); v[i]=v[i-1]i; end; ... plot2d(v,logplot=1);
textbox(latex(log(n)),x=0.5):
```

```
!images/23030630036_Marcelline
```

```
A=[0.5,0.2;0.7,0.1]; b=[2;2]; ... x=[1;1]; repeat xnew=A.x-b; until all(xnew=x);
x=xnew; end; ... x,
-7.09677 -7.74194
```

Iterasi di dalam Fungsi

Fungsi atau program juga dapat menggunakan iterasi dan dapat digunakan untuk melakukan iterasi. Berikut adalah beberapa contoh iterasi di dalam fungsi.

Contoh berikut adalah suatu fungsi untuk menghitung berapa lama suatu iterasi konvergen. Nilai fungsi tersebut adalah hasil akhir iterasi dan banyak iterasi sampai konvergen.

```
function map hiter(f,x0)...
```

```
x=x0; maxiter=0; repeat xnew=f(x); maxiter = maxiter + 1; until xnew =
x; x = xnew; end; return maxiter; endfunction < /pre > Misalnya, berikut adalah iterasi untuk mendapatkan
```

5, jika dimulai dari hampiran awal 2.

```
hiter("(x+2/x)/2",2)
```

5

Karena fungsinya didefinisikan menggunakan "map". maka nilai awalnya dapat berupa vektor.

```
x=1.5:0.1:10; hasil=hiter("(x+2/x)/2",x); ... plot2d(x,hasil):
```

```
!images/23030630036_Marcelline
```

Dari gambar di atas terlihat bahwa kekonvergenan iterasinya semakin lambat, untuk nilai awal semakin besar, namun penambahannya tidak kontinu. Kita dapat menemukan kapan maksimum iterasinya bertambah.

```
hasil[1:10]
[4, 5, 5, 5, 5, 5, 6, 6, 6, 6]
x[nonzeros(differences(hasil))]
[1.5, 2, 3.4, 6.6]
```

maksimum iterasi sampai konvergen meningkat pada saat nilai awalnya 1.5, 2, 3.4, dan 6.6.

Contoh berikutnya adalah metode Newton pada polinomial kompleks berderajat 3.

```
p = x3 - 1; newton = x - p/diff(p, x); newton
```

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

Selanjutnya didefinisikan fungsi untuk melakukan iterasi (aslinya 10 kali).

```
function iterasi(f, x, n = 10)...
loop 1 to n; x=f(x); end; return x; endfunction < /pre > Kitamulaidenganmenentukantitik-
titikgridpadabidangkompleksnya.
```

```
r=1.5; x=linspace(-r,r,501); Z=x+Ix'; W=iterasi(newton,Z);
```

Berikut adalah akar-akar polinomial di atas.

```
z=solve(p)()
```

```
[ -0.5+0.866025i, -0.5-0.866025i, 1+0i ]
```

Untuk menggambar hasil iterasinya, dihitung jarak dari hasil iterasi ke-10 ke masing-masing akar, kemudian digunakan untuk menghitung warna yang akan digambar, yang menunjukkan limit untuk masing-masing nilai awal.

Fungsi `plotrgb()` menggunakan jendela gambar terkini untuk menggambar warna RGB sebagai matriks.

```
C=rgb(max(abs(W-z[1]),1),max(abs(W-z[2]),1),max(abs(W-z[3]),1)); ... plot2d(none,-
r,r,-r,r); plotrgb(C):
```

```
![images/23030630036_Marcelline
```

Iterasi Simbolik

Seperti sudah dibahas sebelumnya, untuk menghasilkan barisan ekspresi simbolik dengan Maxima dapat digunakan fungsi `makelist()`.

`powerdisp:true` // untuk menampilkan deret pangkat mulai dari suku berpangkat terkecil

```
true
```

```
deret = makelist(taylor(exp(x),x,0,k),k,1,3); deret//barisanderetTayloruntukex
```

$$\left[ 1 + x, 1 + x + \frac{x^2}{2}, 1 + x + \frac{x^2}{2} + \frac{x^3}{6} \right]$$

Untuk mengubah barisan deret tersebut menjadi vektor string di EMT digunakan fungsi `mxm2str()`. Selanjutnya, vektor string/ekspresi hasilnya dapat digambar seperti menggambar vektor ekspresi pada EMT.

```
plot2d("exp(x)",0,3); // plot fungsi aslinya, ex
```

```
plot2d(mxm2str("deret"), add,color=4:6): // plot ketiga deret taylor ham-
piran fungsi tersebut
```

```
![images/23030630036_Marcelline
```

Selain cara di atas dapat juga dengan cara menggunakan indeks pada vektor/list yang dihasilkan.

```
deret[3]
```

$$1 + x + \frac{x^2}{2} + \frac{x^3}{6}$$

```
plot2d(["exp(x)",deret[1],deret[2],deret[3]],0,3,color=1:4):
!images/23030630036_Marcelline
sum(sin(kx)/k,k,1,5)
```

$$\sin x + \frac{\sin(2x)}{2} + \frac{\sin(3x)}{3} + \frac{\sin(4x)}{4} + \frac{\sin(5x)}{5}$$

Berikut adalah cara menggambar kurva

$$y = \sin(x) + \sin 3x + \sin 5x + \dots$$

```
plot2d(sum(sin((2k+1)x)/(2k+1),k,0,20),0,2pi):
!images/23030630036_Marcelline
```

Hal serupa juga dapat dilakukan dengan menggunakan matriks, misalkan kita akan menggambar kurva

$$y = \sum_{k=1}^{100} \sin(kx)k, \quad 0 \leq x \leq 2\pi.$$

```
x=linspace(0,2pi,1000); k=1:100; y=sum(sin(kx')/k)'; plot2d(x,y):
!images/23030630036_Marcelline
Tabel Fungsi
```

Terdapat cara menarik untuk menghasilkan barisan dengan ekspresi Maxima. Perintah `mxmtable()` berguna untuk menampilkan dan menggambar barisan dan menghasilkan barisan sebagai vektor kolom.

Sebagai contoh berikut adalah barisan turunan ke- $n$   $x^x dx = 1$ .

```
mxmtable("diffat(x^x,x=1,n)", "n", 1,8,frac=1);
```

```
1 2 3 8 10 54 -42 944
```

```
!images/23030630036_Marcelline
```

```
'sum(k,k,1,n) = factor(ev(sum(k,k,1,n),simpsum=true))//simpsum :
menghitungderetsecarasimbolik
```

$$\sum_{k=1}^n k = \frac{n(1+n)}{2}$$

```
'sum(1/(3^k+k),k,0,inf) = factor(ev(sum(1/(3^k+k),k,0,inf),simpsum=
true))
```

$$\sum_{k=0}^{\infty} \frac{1}{k+3^k} = \sum_{k=0}^{\infty} \frac{1}{k+3^k}$$

Di sini masih gagal, hasilnya tidak dihitung.

```
'sum(1/x^2,x,1,inf) = ev(sum(1/x^2,x,1,inf),simpsum=true)//ev : menghitungnilaieksresi
```

$$\sum_{x=1}^{\infty} \frac{1}{x^2} = \frac{\pi^2}{6}$$

*'sum((-1)<sup>(k-1)/k</sup>, k, 1, inf) = factor(ev(sum((-1)<sup>(x-1)/x</sup>, x, 1, inf), simpsum = true))*

$$\sum_{k=1}^{\infty} \frac{(-1)^{-1+k}}{k} = - \sum_{x=1}^{\infty} \frac{(-1)^x}{x}$$

Di sini masih gagal, hasilnya tidak dihitung.

*'sum((-1)<sup>k/(2k-1)</sup>, k, 1, inf) = factor(ev(sum((-1)<sup>k/(2k-1)</sup>, k, 1, inf), simpsum = true))*

$$\sum_{k=1}^{\infty} \frac{(-1)^k}{-1+2k} = \sum_{k=1}^{\infty} \frac{(-1)^k}{-1+2k}$$

*ev(sum(1/n!, n, 0, inf), simpsum = true)*

$$\sum_{n=0}^{\infty} \frac{1}{n!}$$

Di sini masih gagal, hasilnya tidak dihitung, harusnya hasilnya e.

*assume(abs(x) < 1); 'sum(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.

*'sum(x<sup>k</sup>/k!, k, 0, inf) = ev(sum(x<sup>k</sup>/k!, k, 0, inf), simpsum = true)*

$$\sum_{k=0}^{\infty} \frac{x^k}{k!} = \sum_{k=0}^{\infty} \frac{x^k}{k!}$$

*limit(sum(x<sup>k</sup>/k!, k, 0, n), n, inf)*

$$\lim_{n \rightarrow \infty} \sum_{k=0}^n \frac{x^k}{k!}$$

function d(n) = sum(1/(k<sup>2</sup> - k), k, 2, n); d(n)=d(n)

$$d(n) = \sum_{k=2}^n \frac{1}{-k + k^2}$$

*d(10) = ev(d(10), simpsum = true)*

$$\sum_{k=2}^{10} \frac{1}{-k + k^2} = \frac{9}{10}$$

$d(100) = ev(d(100), simpsum = true)$

$$\sum_{k=2}^{100} \frac{1}{-k + k^2} = \frac{99}{100}$$

Deret Taylor

Deret Taylor suatu fungsi  $f$  yang diferensiabel sampai tak hingga di sekitar  $x=a$  adalah:

$$f(x) = \sum_{k=0}^{\infty} \frac{(x-a)^k f^{(k)}(a)}{k!}.$$

$'e^x = \text{taylor}(\exp(x), x, 0, 10) // \text{deret Taylore}^x \text{ disekitar } x = 0, \text{ sampai sukuke} -$   
11

$$e^x = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{x^5}{120} + \frac{x^6}{720} + \frac{x^7}{5040} + \frac{x^8}{40320} + \frac{x^9}{362880} + \frac{x^{10}}{3628800}$$

$'\log(x) = \text{taylor}(\log(x), x, 1, 10) // \text{deret log}(x) \text{ disekitar } x = 1$

$$\log x = -1 - \frac{(-1+x)^2}{2} + \frac{(-1+x)^3}{3} - \frac{(-1+x)^4}{4} + \frac{(-1+x)^5}{5} - \frac{(-1+x)^6}{6} + \frac{(-1+x)^7}{7} - \frac{(-1+x)^8}{8} + \frac{(-1+x)^9}{9} -$$

## 6 Geometri

23030630036\_MarcellineCalyaPadmarini\_EMTGeometriNama : MarcellineCalyaPadmarini

NIM : 23030630036

Kelas : Matematika E

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 setPlorange(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 lt;ABC  
 angleBisector(A, B, C): garis bagi sudut lt;ABC  
 circleWithCenter (A, r): lingkaran dengan pusat A dan jari-jari r  
 getCircleCenter(c): pusat lingkaran c  
 getCircleRadius(c): jari-jari lingkaran c  
 circleThrough(A,B,C): lingkaran melalui A, B, C  
 middlePerpendicular(A, B): titik tengah AB  
 lineCircleIntersections(g, c): titik potong garis g dan lingkaran c  
 circleCircleIntersections (c1, c2): titik potong lingkaran c1 dan c2  
 planeThrough(A, B, C): bidang melalui titik A, B, C  
 Fungsi-fungsi Khusus Untuk Geometri Simbolik:  
 getLineEquation (g,x,y): persamaan garis g dinyatakan dalam x dan y getHesseForm (g,x,y,A): bentuk Hesse garis g dinyatakan dalam x dan y dengan titik A pada sisi positif (kanan/atas) garis  
 quad(A,B): kuadrat jarak AB  
 spread(a,b,c): Spread segitiga dengan panjang sisi-sisi a,b,c, yakni  $\sin(\alpha)^2$  dengan  $\alpha$  sudut yang menghadap sisi a.

`crosslaw(a,b,c,sa)`: persamaan 3 quads dan 1 spread pada segitiga dengan panjang sisi a, b, c.  
`triplespread(sa,sb,sc)`: persamaan 3 spread sa,sb,sc yang memebntuk suatu segitiga  
`doublespread(sa)`: Spread sudut rangkap Spread  $2\phi$ , dengan  $sa = \sin(\phi)^2 spreada$ .  
 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 garis  
`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  
 Grafiknya  
`plotPoint(D,value=1)`; // koordinat D ditampilkan  
`aspect(1); plotSegment(A,D)`; // tampilkan semua gambar hasil plot...()  
 ![images/23030630036\_Marcelline  
 Hitung luas ABC:

$$L_{\triangle ABC} = \frac{1}{2}AD.BC.$$

`norm(A-D)norm(B-C)/2` // AD=norm(A-D), BC=norm(B-C)

1.5

Bandingkan dengan rumus determinan.

`areaTriangle(A,B,C)` // hitung luas segitiga langsung dengan fungsi

1.5

Cara lain menghitung luas segitigas ABC:

`distance(A,D)distance(B,C)/2`

1.5

Sudut berada pada e

`degprint(computeAngle(B,C,A))`



36°52'11.63"

Maka keliling lingkaran luar segitiga adalah

```
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/23030630036_Marcelline
```

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 jACB
```

```
g=angleBisector(C,A,B); // garis bagi jCAB
```

```
P=lineIntersection(l,g) // titik potong kedua garis bagi sudut
```

```
[0.86038, 0.86038]
```

tambahkan semuanya ke plot

```
color(5); plotLine(l); plotLine(g); color(1); // gambar kedua garis bagi sudut
```

```
plotPoint(P,"P"); // gambar titik potongnya
```

```
r=norm(P-projectToLine(P,lineThrough(A,B))) // jari-jari lingkaran dalam  
0.509653732104
```

```
plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC"); // gambar lingkaran dalam
```

```
![images/23030630036_Marcelline
```

Latihan

1. Tentukan ketiga titik singgung lingkaran dalam dengan sisi-sisi segitiga ABC.

2. Gambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut. Merupakan segitiga apakah itu?

3. Hitung luas segitiga tersebut.

4. Tunjukkan bahwa garis bagi sudut yang ke tiga juga melalui titik pusat lingkaran dalam.

5. Gambar jari-jari lingkaran dalam.

6. Hitung luas lingkaran luar dan luas lingkaran dalam segitiga ABC. Adakah hubungan antara luas kedua lingkaran tersebut dengan luas segitiga ABC?

```
reset;
```

```
plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga ABC");
```

```
K = lineCircleIntersections(lineThrough(A,B),circleWithCenter(P,r))
```

```
[0.5, 0.5]
```

```
L = lineCircleIntersections(lineThrough(C,B),circleWithCenter(P,r))
```

```
[0.632456, 1.31623]
```

```
M = lineCircleIntersections(lineThrough(C,A),circleWithCenter(P,r))
```

```
[1.31623, 0.632456]
```

```
plotPoint(K,"K"); plotPoint(M,"M"); plotPoint(L,"L");
```

```
!images/23030630036_Marcelline
```

```
plotSegment(K,L,"m");
```

```
plotSegment(L,M,"n");
```

```
plotSegment(M,K,"l");
```

```
!images/23030630036_Marcelline
```

```
distance(P,lineIntersection(lineThrough(A,B),l))
```

```
0.509653732104
```

```
getCircleRadius(circleWithCenter(P,r))
```

```
0.509653732104
```

Contoh 2: Geometri Smbolik

Kita dapat menghitung geometri eksak dan simbolik menggunakan Maxima.

File geometri.e menyediakan fungsi yang sama (dan lebih banyak lagi) di Maxima. Namun, kita dapat menggunakan perhitungan simbolik sekarang.

A = [1,0]; B = [0,1]; C = [2,2]; // menentukan tiga titik A, B, C

Fungsi untuk garis dan lingkaran bekerja seperti fungsi Euler, tetapi 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/23030630036_Marcelline
```

```
getLineEquation(lineThrough([x1,y1],[x2,y2]),x,y),solve(
```

$$\left[ y = \frac{-(x_1 - x) y_2 - (x - x_2) y_1}{x_2 - x_1} \right]$$

```
!images/23030630036_Marcelline
```

```
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//(*titikpusatdanjari-jari*)*lingkaranmelaluiA, 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/23030630036\_Marcelline  
computeAngle(A, C, B)//nilai < ACB

$$\arccos\left(\frac{4}{5}\right)$$

solve(getLineEquation(angleBisector(A, C, B), x, y), y)[1]//persamaan garis bagi < ACB

$$y = x$$

P = lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A)); P//*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]

Berpotongan Garis dan Lingkaran

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, f

[4.64575, -1.64575] [-0.645751, 3.64575] 2

plotPoint(P1); plotPoint(P2):

![images/23030630036\_Marcelline

maka sama pada maxima

c = circleWithCenter(A,4) // lingkaran dengan pusat A jari-jari 4

[1, 0, 4]

l = lineThrough(B,C) // garis l melalui B dan C

[1, 1, 3]

lineCircleIntersections(l,c)|radcan, //titikpotonglingkarancdangaris l

$$\left[ \left[ \sqrt{7}+2, 1-\sqrt{7} \right], \left[ 2-\sqrt{7}, \sqrt{7}+1 \right] \right]$$

Akan ditunjukkan bahwa sudut-sudut yang menghadap busur yang sama adalah sama besar.

```
C=A+normalize([-2,-3])4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);
degprint(computeAngle(P1,C,P2))
69°17'42.68"
```

```
C=A+normalize([-4,-3])4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);
degprint(computeAngle(P1,C,P2))
69°17'42.68"
```

```
insimg;
![images/23030630036_Marcelline
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/23030630036_Marcelline
```

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)
```

$$\left[ y = \frac{(b_2 - a_2) x - a_1 b_2 + a_2 b_1}{b_1 - a_1} \right]$$

Perhatikan hasil kali gradien garis g dan h adalah:

$$\frac{-(b_1 - a_1)}{(b_2 - a_2)} \times \frac{(b_2 - a_2)}{(b_1 - a_1)} = -1.$$

Artinya kedua garis tegak lurus.

Contoh 3: Rumus Heron

Rumus Heron menyatakan bahwa luas segitiga dengan panjang sisi-sisi a, b dan c adalah:

$$L = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{dengan } s = (a+b+c)/2,$$

atau bisa ditulis dalam bentuk lain:

$$L = \frac{1}{4} \sqrt{(a+b+c)(b+c-a)(a+c-b)(a+b-c)}$$

Untuk membuktikan hal ini kita misalkan C(0,0), B(a,0) dan A(x,y), b=AC, c=AB. Luas segitiga ABC adalah

$$L_{\triangle ABC} = \frac{1}{2} a \times y.$$

Nilai y didapat dengan menyelesaikan sistem persamaan:

$$x^2 + y^2 = b^2, \quad (x-a)^2 + y^2 = c^2.$$

```
setPlotRange(-1,10,-1,8); plotPoint([0,0], "C(0,0)"); plotPoint([5.5,0], "B(a,0)");
... plotPoint([7.5,6], "A(x,y)");
    plotSegment([0,0],[5.5,0], "a",25); plotSegment([5.5,0],[7.5,6], "c",15); ...
plotSegment([0,0],[7.5,6], "b",25);
    plotSegment([7.5,6],[7.5,0], "t=y",25);
    ![images/23030630036_Marcelline
    assume(a 0); sol = solve([x^2 + y^2 = b^2, (x-a)^2 + y^2 = c^2], [x, y])
    2 2 2 - c + b + a [[x = -----, y = 2 a 4 2 2 2 4 2 2 4 sqrt(- c + 2 b c
+ 2 a c - b + 2 a b - a ) - -----], 2 a 2 2 2 - c
+ b + a [x = -----, y = 2 a 4 2 2 2 4 2 2 4 sqrt(- c + 2 b c + 2 a c - b
+ 2 a b - a ) -----]] 2 a
    Ekstrak solusi y
    ysol = y with sol[2][2]; 'ysol
```

$$\frac{\sqrt{-c^4 + 2b^2c^2 + 2a^2c^2 - b^4 + 2a^2b^2 - a^4}}{2a}$$

kita dapat melihat rumus heron

$$\text{function H(a,b,c) = sqrt(factor((ysola/2)^2)); 'H(a,b,c)=H(a,b,c)}$$

$$H(a, b, c) = \frac{\sqrt{(-c+b+a)(c-b+a)(c+b-a)(c+b+a)}}{4}$$

'Luas = H(2, 5, 6) // luas segitiga dengan panjang sisi - sisi 2, 5, 6

$$Luas = \frac{3\sqrt{39}}{4}$$

Tentu saja, setiap segitiga persegi panjang adalah kasus yang terkenal.

H(3,4,5) // luas segitiga siku-siku dengan panjang sisi 3, 4, 5

6

Dan 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 dengan panjang sisi 3, 4, x (1 ≤ x ≤ 7)

![images/23030630036\_Marcelline

Kasus umum juga berfungsi

$\text{solve}(\text{diff}(H(a, b, c)^2, c) = 0, c)$

$$\left[ c = -\sqrt{b^2 + a^2}, c = \sqrt{b^2 + a^2}, c = 0 \right]$$

Sekarang mari kita cari himpunan semua titik di mana  $b+c=d$  untuk beberapa konstanta  $d$ . Diketahui bahwa ini adalah elips.

$s1 = \text{subst}(d-c, b, \text{sol}[2]); s1$

$$\left[ x = \frac{(d-c)^2 - c^2 + a^2}{2a}, y = \frac{\sqrt{-(d-c)^4 + 2c^2(d-c)^2 + 2a^2(d-c)^2 - c^4 + 2a^2c^2 - a^4}}{2a} \right]$$

dan buat fungsi ini

$\text{function fx}(a, c, d) = \text{rhs}(s1[1]); \text{fx}(a, c, d), \text{function fy}(a, c, d) = \text{rhs}(s1[2]); \text{fy}(a, c, d)$

$$\frac{\sqrt{-(d-c)^4 + 2c^2(d-c)^2 + 2a^2(d-c)^2 - c^4 + 2a^2c^2 - a^4}}{2a}$$

![images/23030630036\_Marcelline

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/23030630036\_Marcelline

Kita dapat memeriksa persamaan umum untuk elips ini, yaitu

$$\frac{(x - x_m)^2}{u^2} + \frac{(y - y_m)^2}{v^2} = 1,$$

dimana  $(x_m, y_m)$  adalah pusat, dan  $u$  dan  $v$  adalah setengah sumbu.

$\text{ratsimp}((\text{fx}(a, c, d) - a/2)^2/u^2 + \text{fy}(a, c, d)^2/v^2 \text{ with } [u = d/2, v = \text{sqrt}(d^2 - a^2)/2])$

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.

$eqns = [\text{diff}(H(a,b,d-(a+b))^2, a) = 0, \text{diff}(H(a,b,d-(a+b))^2, b) = 0]; eqns$

$$\left[ 2H(a,b,d-b-a) \left( \frac{d}{da} H(a,b,d-b-a) \right) = 0, 2H(a,b,d-b-a) \left( \frac{d}{db} H(a,b,d-b-a) \right) = 0 \right]$$

Kami mendapatkan beberapa minima, yang termasuk dalam segitiga dengan satu sisi 0, dan solusinya  $a=b=c=d/3$ .

$\text{solve}(eqns, [a, b])$

$$\left[ \left[ a = \frac{d}{3}, b = \frac{d}{3} \right], \left[ a = 0, b = \frac{d}{2} \right], \left[ a = \frac{d}{2}, b = 0 \right], \left[ a = \frac{d}{2}, b = \frac{d}{2} \right] \right]$$

Ada juga metode Lagrange, memaksimalkan  $H(a,b,c)^2$  terhadap  $a+b+d=d$ .  
to  $a+b+d=d$ .

$\text{solve}([\text{diff}(H(a,b,c)^2, a) = la, \text{diff}(H(a,b,c)^2, b) = la, \dots \text{diff}(H(a,b,c)^2, c) = la, a+b+c=d], [a, b, c, la])$

$\text{d d d d } [[a = 0, b = -, c = -, la = 0], [a = -, b = 0, c = -, la = 0], [a = -, b = -, c = 0, la = 0], [a = -, b = -, c = -, la = -]]$   
108

Kita dapat mengatur plot situasinya

Pertama-tama atur poin di Maxima.

$A = \text{at}([x,y], \text{sol}[2]); A$

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

$B = [0,0]; B, C = [a,0]; C$

$[a,0]$

!images/23030630036\_Marcelline

Kemudian atur rentang plot, dan plot titik-titiknya.

$\text{setPlotRange}(0,5,-2,3); \dots a=4; b=3; c=2; \dots \text{plotPoint}(\text{mxmeval}("B"), "B");$   
 $\text{plotPoint}(\text{mxmeval}("C"), "C"); \dots \text{plotPoint}(\text{mxmeval}("A"), "A");$

!images/23030630036\_Marcelline

Plot segmen

$\text{plotSegment}(\text{mxmeval}("A"), \text{mxmeval}("C")); \dots \text{plotSegment}(\text{mxmeval}("B"), \text{mxmeval}("C"));$   
 $\dots \text{plotSegment}(\text{mxmeval}("B"), \text{mxmeval}("A"));$

!images/23030630036\_Marcelline

Hitung tegak lurus tengah di Maxima

```

h = middlePerpendicular(A,B); g = middlePerpendicular(B,C);
dan pusat lingkaran.
U = lineIntersection(h,g);
Kami mendapatkan rumus untuk jari-jari lingkaran.
assume(a 0,b 0,c 0); distance(U,B)|radcan

```

1

Mari kita dapatkan rumus untuk jari-jari lingkaran

```

plotPoint(U()); ... plotCircle(circleWithCenter(mxmeval("U"),mxmeval("distance(U,C)"))):
! [images/23030630036_Marcelline
Menggunakan geometri, kita dapat memperoleh rumus sederhana

```

$$\frac{a}{\sin(\alpha)} = 2r$$

untuk radiusnya. Kita dapat memeriksa apa ini benar dengan Maxima. Maxima akan memfaktorkan ini jika kita kuadratkan.

```

c^2/sin(computeAngle(A,B,C))^2|factor

```

$$2c^2$$

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.

```

setPlotRange(3); plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C");
Kita juga bisa menambahkan sisi segitiga.
plotSegment(A,B,""); plotSegment(B,C,""); plotSegment(C,A,"");
! [images/23030630036_Marcelline

```

Berikut adalah luas segitiga, menggunakan rumus determinan. Tentu saja, kita harus mengambil nilai absolut dari hasil ini.

```

areaTriangle(A,B,C)

```

$$-\frac{7}{2}$$

Kita dapat menghitung koefisien sisi c

```

c = lineThrough(A,B)
[- 1, 3, - 2]

```



Dan juga dapatkan rumus untuk baris ini

*getLineEquation(c, x, y)*

$$3y - x = -2$$

Untuk bentuk Hesse, kita perlu menentukan sebuah titik, sehingga titik tersebut berada di sisi positif dari bentuk Hesse. Memasukkan titik menghasilkan jarak positif ke garis.

*getHesseForm(c, x, y, C), at(*

$$\frac{7}{\sqrt{10}}$$

! [images/23030630036\_Marcelline

Sekarang kita hitung keliling 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/23030630036\_Marcelline

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 kita

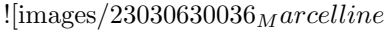
plotPoint(H(), "H"); plotLine(el(), "Garis Euler");

! [images/23030630036\_Marcelline

Pusat gravitasi harus berada di garis ini.

M = (A+B+C)/3; *getLineEquation(el, x, y) with [x = M[1], y = M[2]]*

$$-\frac{1}{2} = -\frac{1}{2}$$

plotPoint(M(),"M"): // titik berat  
  
 Teorinya 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 pusat dalam lingkaran kurang bagus

$Q = \text{lineIntersection}(\text{angleBisector}(A, C, B), \text{angleBisector}(C, B, A))$ —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

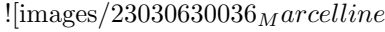
$r = \text{distance}(Q, \text{projectToLine}(Q, \text{lineThrough}(A, B)))$ —ratsimp;  $r$

$$\frac{\sqrt{(-41\sqrt{2} - 31) \sqrt{5} \sqrt{13} + 115\sqrt{2} + 614}}{7\sqrt{2}}$$

$LD = \text{circleWithCenter}(Q, r)$ ; // Lingkaran dalam

Mari kita tambahkan ini ke plot.

$\text{color}(5); \text{plotCircle}(LD());$



Parabola

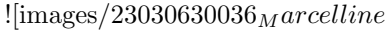
Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

$p = \text{getHesseForm}(\text{lineThrough}(A, B), x, y, C) - \text{distance}([x, y], C); p = '0$

$$\frac{3y - x + 2}{\sqrt{10}} - \sqrt{(2 - y)^2 + (1 - x)^2} = 0$$

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

$\text{plot2d}(p, \text{level}=0, \text{add}=1, \text{contourcolor}=6);$



Ini seharusnya menjadi beberapa fungsi, tetapi pemecah default Maxima hanya dapat menemukan solusinya, jika kita kuadratkan persamaannya. Akibatnya, kami mendapatkan solusi palsu.

```
akar = solve(getHesseForm(lineThrough(A,B),x,y,C)^2-distance([x,y],C)^2,y)
[y = - 3 x - sqrt(70) sqrt(9 - 2 x) + 26, y = - 3 x + sqrt(70) sqrt(9 - 2 x) +
26]
```

Solusi pertama adalah

$$y = -3x - \sqrt{70}\sqrt{9-2x} + 26$$

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/23030630036_Marcelline
function g(x) = rhs(akar[1]); 'g(x) = g(x)//fungsiyangmende finisikankurvadiatas
```

$$g(x) = -3x - \sqrt{70}\sqrt{9-2x} + 26$$

```
T = [-1, g(-1)]; // ambil sebarang titik pada kurva tersebut
dT C = distance(T,C); fullratsimp(dTC),float(
```

2.135605779339061

```
![images/23030630036_Marcelline
```

```
U = projectToLine(T,lineThrough(A,B)); U//proyeksiTpadagarisAB
```

$$\left[ \frac{80 - 3\sqrt{11}\sqrt{70}}{10}, \frac{20 - \sqrt{11}\sqrt{70}}{10} \right]$$

```
dU2AB = distance(T,U); fullratsimp(dU2AB),float(
```

2.135605779339061

```
![images/23030630036_Marcelline
```

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/23030630036\_Marcelline

Tentu saja,

$$\sin(w_a) = \frac{a}{c},$$

di mana  $w_a$  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 kuadrat jarak. 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  $w_a$  menjadi sprad, dan mencetaknya sebagai pecahan.

aran 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.

$$\frac{\sin(wa)^2}{9/25}$$

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  $s_a$  adalah penyebaran sudut A. Sisi a, seperti biasa, berhadapan dengan sudut A.

Hukum ini diimplementasikan dalam file geometri.e yang kami muat ke Euler.

$$\text{crosslaw}(aa, bb, cc, saa)$$

$$\left[ \left( bb - aa + \frac{7}{6} \right)^2, \left( bb - aa + \frac{7}{6} \right)^2, \left( bb - aa + \frac{5}{3\sqrt{2}} \right)^2 \right] = \left[ \frac{14 bb (1 - saa)}{3}, \frac{14 bb (1 - saa)}{3}, \frac{5 \cdot 2^{\frac{3}{2}} bb (1 - saa)}{3} \right]$$

Dalam kasus ini, kita dapatkan

$$\text{crosslaw}(a, b, c, sa)$$

$$1024 = 1024$$

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  $s_a$ .

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}(\$$

$$\left[ x = \frac{9}{25} \right]$$

!images/23030630036\_Marcelline

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.

$$\text{solve}(\text{crosslaw}(aa, bb, cc, x), x)$$

$$\left[ \left[ \frac{168 bb x + 36 bb^2 + (-72 aa - 84) bb + 36 aa^2 - 84 aa + 49}{36}, \frac{168 bb x + 36 bb^2 + (-72 aa - 84) bb + 36 aa^2 - 84 aa + 49}{36} \right] \right]$$

We could make a function of the result. Such a function is already defined in the geometry.e file of Euler.

*spread(a, b, c)*

$$\frac{9}{25}$$

Sebagai contoh, kita dapat menggunakannya untuk menghitung sudut segitiga dengan sisi

$$a, \quad a, \quad \frac{4a}{7}$$

Hasilnya rasional yang tidak begitu mudah didapat jika kita menggunakan trigometri klasik.

*spread(a, a, 4a/7)*

$$\frac{6}{7}$$

Ini adalah sudut dalam derajat.

nakannya untuk menghitung sudut segitiga dengan sisi

$$a, \quad a, \quad \frac{4a}{7}$$

Hasilnya rasional yang tidak begitu mudah didapat jika kita menggunakan trigometri klasik.

*degprint(arcsin(sqrt(6/7)))*

67°47'32.44"

Contoh Lainnya

Sekarang, mari kita coba conoh 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"); ... insing;

![images/23030630036\_Marcelline

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,*

![[images/23030630036\_Marcelline

![[images/23030630036\_Marcelline

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/23030630036\_Marcelline

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  $h_a$  di sisi a. Ingat bahwa

$$s_b = \frac{h_a}{c}$$

dari definisi.

$$h_a = c s_b; h_a$$

$$\frac{49}{17}$$

Gambar berikut telah dihasilkan dengan program geometri C.a.R., yang dapat menggambar kuadrat dan menyebar.

image: (20) RationalGeometryCaR.png

Berdasarkan definisi, panjang  $h_a$  adalah akar kuadrat dari kuadratnya.

$$\sqrt{h_a}$$

$$\frac{7}{\sqrt{17}}$$

Sekarang kita dapat menghitung luas segitiga. Jangan lupa, bahwa kita berhadapan dengan kuadrat!

$$\sqrt{h_a}\sqrt{a}/2$$

$$\frac{7}{2}$$

Rumus determinan biasa menghasilkan hasil yang sama.

$$\text{areaTriangle}(B, A, C)$$

$$\frac{7}{2}$$

Rumus Heron

Sekarang, kita selesaikan masalah ini seperti biasanya!

$$\text{remvalue}(a, b, c, s_b, h_a);$$

Pertama kita hitung spread di B untuk segitiga dengan sisi  $a$ ,  $b$ , dan  $c$ . Kemudian kita hitung luas kuadrat ("quadrea?"), faktorkan dengan Maxima, dan kita akan mendapatkan rumus Heron yang terkenal.

Memang, hal ini sulit dilakukan dengan pensil dan kertas.

$$\text{spread}(b^2, c^2, a^2), \text{factor}(\$$

$$\frac{(-c + b + a)(c - b + a)(c + b - a)(c + b + a)}{16}$$

![images/23030630036\_Marcelline

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.

$$\text{remvalue}(s_a, s_b, s_c); \text{triplespread}(s_a, s_b, s_c)$$

$$(s_c + s_b + s_a)^2 = 2(s_c^2 + s_b^2 + s_a^2) + 4s_a s_b s_c$$

Aturan ini berlaku untuk setiap tiga sudut yang menambah ke  $180^\circ$ .

$$\alpha + \beta + \gamma = \pi$$



Sejak menyebarkan

$$\alpha, \pi - \alpha$$

adalah sama, aturan triple spread juga benar. Jika

$$\alpha + \beta = \gamma$$

karena penyebaran sudut negatif sama, aturan triple spread juga berlaku, jika

$$\alpha + \beta + \gamma = 0$$

Sebagai contoh, kita dapat menghitung penyebaran sudut  $60^\circ$ . Ini  $3/4$ . Persamaan memiliki dua solusi, bagaimanapun semua spread adalah 0.

*solve(triplespread(x, x, x), x)*

$$\left[ x = \frac{3}{4}, x = 0 \right]$$

Sebaran  $90^\circ$  dengan jelas 1. Jika dua sudut dijumlahkan menjadi  $90^\circ$ , sebarannya menyelesaikan persamaan rangkap tiga dengan a,b,1. Dengan perhitungan berikut kita mendapatkan a+b=1.

*triplespread(x, y, 1), solve(*

$$[x = 1 - y]$$

!images/23030630036\_Marcelline

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.

*solve(triplespread(a, a, x), x), functiondoublespread(a) = factor(rhs(*

$$[x = 4a - 4a^2, x = 0]$$

- 4 (a - 1) a

Sudut Bagi

Ini adalah 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/23030630036\_Marcelline

Mari kita hitung panjang garis bagi sudut di A. Tetapi kita ingin menyelesaikannya untuk umum a,b,c.

remvalue(a,b,c);

Pertama-tama kita hitung penyebaran sudut yang dibagi dua di A, menggunakan rumus triple spread.

Masalah dengan rumus ni muncul lagi. Ini memiliki dua solusi. Kita harus memilih yang benar. Solusi lainnya mengacu pada sudut terbelah  $180^\circ$ -wa.

$triplespread(x, x, a/(a+b)), solve($

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

![images/23030630036\_Marcelline

![images/23030630036\_Marcelline

Mari kita periksa persegi panjang Egyptian.

$sa2with[a = 3^2, b = 4^2]$

$$\frac{1}{10}$$

Kita dapat mencetak sudut dalam Euler, setelah mentransfer penyebaran ke radian.

$wa2 := \arcsin(\sqrt{1/10}); \text{degprint}(wa2)$

18°26'5.82"

Titik P adalah perpotongan garis bagi sudut dengan sumbu y.

e y-axis.

$P := [0, \tan(wa2)4]$

[0, 1.33333]

$\text{plotPoint}(P, "P"); \text{plotSegment}(A, P):$

![images/23030630036\_Marcelline

Mari kita periksa sudut dalam contoh spesifik kita.

$\text{computeAngle}(C, A, P), \text{computeAngle}(P, A, B)$

0.321750554397 0.321750554397

Sekarang kita hitung panjang dari garis bagi AP.

Kita menggunakan teorema sinus dalam segitiga APC. Teorema ini menyatakan bahwa

$$\frac{BC}{\sin(w_a)} = \frac{AC}{\sin(w_b)} = \frac{AB}{\sin(w_c)}$$

berlaku dalam segitiga apapun. Kuadratkan itu diterjemahkan ke dalam apa yang disebut "hukum penyebaran"

$$\frac{a}{s_a} = \frac{b}{s_b} = \frac{c}{s_b}$$

dimana a, b, c menunjukkan kuadran.

Karena spread CPA adalah  $1-sa^2$ , kita dapatkan darinya  $bisa/1=b/(1-sa^2)$  dan menghitung bisa (kuadran dari garis bagi sudut).

$\text{factor}(\text{ratsimp}(b/(1-sa^2))); \text{bisa} =$

$$\frac{2b(b+a)}{\sqrt{b(b+a)} + b + a}$$

Mari kita periksa rumus ini untuk nilai-nilai Egyptian

$\sqrt{\text{mxmeval}("at(bisa, [a=3^2, b=4^2])")}, \text{distance}(A, P)$

4.21637021356 4.21637021356

Kita juga dapat menghitung P menggunakan rumus spread.

```
py=factor(ratsimp(sa2bisa)); py
```

$$-\frac{b \left( \sqrt{b(b+a)} - b - a \right)}{\sqrt{b(b+a)} + b + a}$$

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/23030630036_Marcelline
```

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 bisa membuat fungsi euler.

```
function periradius(a,b,c) = rabc;
mari kita cek hasil 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)rabc|ratsimp
```

$$\frac{b}{4}$$

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/23030630036\_Marcelline

dan grafiknya juga agak sederhana: bagian atar kerucut.

```
plot3d("d1",xmin=-2,xmax=0,ymin=-2,ymax=0):
```

![images/23030630036\_Marcelline

Tentu saja titik minimum 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/23030630036\_Marcelline

Grafiknya menjadi lebih menarik

```
plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
```

![images/23030630036\_Marcelline

Pembatasan garis (AB) lebih terkenal

```
plot2d("abs(x+1)+abs(x-1)",xmin=-3,xmax=3):
```

![images/23030630036\_Marcelline

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/23030630036\_Marcelline

```
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/23030630036\_Marcelline

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/23030630036_Marcelline
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/23030630036_Marcelline
```

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):
!images/23030630036_Marcelline
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/23030630036_Marcelline
```

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:

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!

```
C=[-1,1];
plot3d("d4",xmin=-1,xmax=1,ymin=-1,ymax=1);
![images/23030630036_Marcelline
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;
![images/23030630036_Marcelline
```

Contoh 7: Bola Dandelin dengan Povray

Anda dapat menjalankan demonstrasi ini, jika Anda telah menginstal Povray, dan pengine.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 geometry.e dari Euler untuk ini.

```
load geometry;
```

Pertama dua garis yang membentuk kerucut.

```
g1 = lineThrough([0,0],[1,a])
[- a, 1, 0]
g2 = lineThrough([0,0],[-1,a])
[- a, - 1, 0]
```

Kemudian 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/23030630036_Marcelline
```

Sekarang kita ambil titik umum pada sumbu-y.

```
P = [0,u]
[0, u]
```

Hitung jarak ke g1.

$d1 = \text{distance}(P, \text{projectToLine}(P, g1)); d1$

$$\sqrt{\left(\frac{a^2 u}{a^2 + 1} - u\right)^2 + \frac{a^2 u^2}{(a^2 + 1)^2}}$$

Hitung jarak ke g

$d = \text{distance}(P, \text{projectToLine}(P, g)); d$

$$\sqrt{\left(\frac{u+2}{5} - u\right)^2 + \frac{(2u-1)^2}{25}}$$

Dan temukan pusat kedua lingkaran yang jaraknya sama.

$\text{sol} = \text{solve}(d1^2 = d^2, u); \text{sol}$

$$\left[ u = \frac{-\sqrt{5}\sqrt{a^2+1} + 2a^2 + 2}{4a^2 - 1}, u = \frac{\sqrt{5}\sqrt{a^2+1} + 2a^2 + 2}{4a^2 - 1} \right]$$

Ada dua solusi.

Kami mengevaluasi solusi simbolis, dan menemukan kedua pusat, dan kedua jarak.

$u := \text{sol}()$

$[0.333333, 1]$

$dd := d()$

$[0.149071, 0.447214]$

Plot lingkaran ke dalam gambar.

$\text{color}(\text{red});$

$\text{plotCircle}(\text{circleWithCenter}([0, u[1]], dd[1]), "");$

$\text{plotCircle}(\text{circleWithCenter}([0, u[2]], dd[2]), "");$

$\text{insimg};$

$![\text{images}/23030630036_{\text{Marcelline}}$

Plot dengan Povray

Selanjutnya kami merencanakan semuanya dengan Povray. Perhatikan bahwa Anda mengubah perintah apapun dalam urutan perintah Povray berikut, dan menjalankan kembali semua perintah dengan Shift-Return.

Pertama kita memuat fungsi povray

$\text{load povray};$

$\text{defaultpovray} = "C:"$

Program Files

POV-Ray

v3.7

bin

pvenGINE.exe"

$C:\text{Files-Ray}3.7.\text{exe}$

Kita atur adegan dengan tepat

$\text{povstart}(\text{zoom}=11, \text{center}=[0, 0, 0.5], \text{height}=10^\circ, \text{angle}=140^\circ);$

Kemudian 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 menyen-
```

tuh 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)));
Kemudian kita hitung perpotongan P1P2 dengan bidang.
t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-t2)(P2-P1);
writeln(povpoint(P5,povlook(yellow)));
Kita menghubungkan titik-titik dengan segmen garis.
writeln(povsegment(P1,P2,povlook(yellow)));
writeln(povsegment(P5,P3,povlook(yellow)));
writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang kita hasilkan pita abu-abu, dimana 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/23030630036_Marcelline
```

To get an Anaglyph of this we need to put everything into a scene function.  
This function will be used twice later.

```
function scene () ...
global a,u,dd,g,g1,defaultpointsize; writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));
writeln(povsphere([0,0,u[2]],dd[2],povlook(red))); writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)));
gp=g(); pc=povcone([0,0,0],0,[0,0,a],1,""); vp=[gp[1],0,gp[2]]; dp=gp[3]; writeln(povplane(vp,dp,povlook(blue,0.5),pc));
P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]); writeln(povpoint(P1,povlook(yellow)));
P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]); writeln(povpoint(P2,povlook(yellow)));
```



```

P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]]; writeln(povpoint(P3,povlook(yellow)));
P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]]; writeln(povpoint(P4,povlook(yellow)));
t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-t2)*(P2-P1); writeln(povpoint(P5,povlook(yellow)));
writeln(povsegment(P1,P2,povlook(yellow))); writeln(povsegment(P5,P3,povlook(yellow)));
writeln(povsegment(P5,P4,povlook(yellow))); pcw=povcone([0,0,0],0,[0,0,a],1.01);
pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2],1);
writeln(povintersection([pcw,pc1],povlook(gray))); pc2=povcylinder([0,0,P2[3]-
defaultpointsize/2],[0,0,P2[3]+defaultpointsize/2],1); writeln(povintersection([pcw,pc2],povlook(gray)));
endfunction ;/prej. Anda membutuhkan kacamata merah/cyan untuk menghar-
gai efek berikut.

```

```

    povanaglyph("scene",zoom=11,center=[0,0,0.5],height=10°,angle=140°);
    ![images/23030630036_Marcelline

```

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.

```
the 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.

```
posisi spherical)
```

```
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
```

Ada fungsi untuk heading, dengan mempertimbangkan bentuk elips bumi. Sekali lagi, kami mencetak dengan cara yang cangguh.

```
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

```

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.

```

    pada garis lintang 7°.
    esarea(Solo,FMIPA,Semarang)- " km2", //perkiraanyangsamadenganfungsiesarea()
    2123.64310526 km2

```

Kita juga dapat menambahkan vektor ke posisi. Sebuah vektor berisi heading dan jarak, keduanya dalam radian. Untuk mendapatkan sebuah vektor, kita menggunakan vektor. Untuk menambahkan vektor ke posisi, kami menggunakan saddvektor.

```

    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
    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 dengan 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/23030630036\_Marcelline

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/23030630036\_Marcelline

Latihan

1. Gambarlah segi-n beraturan jika diketahui titik pusat O, n, dan jarak titik pusat ke titik-titik sudut segi-n tersebut (jari-jari lingkaran luar segi-n), r.

Petunjuk:

\* Besar sudut pusat yang menghadap masing-masing sisi segi-n adalah  $\frac{360}{n}$ .

\* 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.

//segitiga

```
setPlotRange(-2.4,2.4,-2.4,2.4); ... A=[-1,-1]; plotPoint(A,"A"); ... B=[1,-1]; plotPoint(B,"B"); ... C=[0,2]; plotPoint(C,"C"); ... plotSegment(A,B,"c"); ... plotSegment(B,C,"a"); ... plotSegment(A,C,"b"); ... aspect(1):
```

!images/23030630036\_Marcelline

```
c=circleThrough(A,B,C); R=getCircleRadius(c); ... O=getCircleCenter(c); ... plotPoint(O,"O"); ... l=angleBisector(A,C,B); ... color(2); plotLine(1); color(1); ... plotCircle(c,"Lingkaran luar segitiga ABC");
```

!images/23030630036\_Marcelline

3. Gambarlah suatu segi-4 yang diketahui keempat titik sudutnya, misalnya A, B, C, D.

- Tentukan apakah segi-4 tersebut merupakan segi-4 garis singgung (sisinya-sisinya merupakan garis singgung lingkaran yang sama yakni lingkaran dalam segi-4 tersebut).

- Suatu segi-4 merupakan segi-4 garis singgung apabila keempat garis bagi sudutnya bertemu di satu titik.

- Jika segi-4 tersebut merupakan segi-4 garis singgung, gambar lingkaran dalamnya.

- Tunjukkan bahwa syarat suatu segi-4 merupakan segi-4 garis singgung apabila hasil kali panjang sisi-sisi yang berhadapan sama.

```
setPlotRange(-4.5,4.5,-4.5,4.5); ... A=[-2,-2]; plotPoint(A,"A"); ... B=[2,-2]; plotPoint(B,"B"); ... C=[2,2]; plotPoint(C,"C"); ... D=[-2,2]; plotPoint(D,"D"); ... plotSegment(A,B,""); ... plotSegment(B,C,""); ... plotSegment(C,D,""); ... plotSegment(A,D,""); ... aspect(1):
```

!images/23030630036\_Marcelline

```
l=angleBisector(A,B,C); ... m=angleBisector(B,C,D); ... P=lineIntersection(l,m); ... color(5); plotLine(1); plotLine(m); color(1); ... plotPoint(P,"P");
```

!images/23030630036\_Marcelline

Dari gambar diatas, dapat dilihat bahwa keempat garis 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/23030630036\_Marcelline

Dari gambar diatas bahwa pada sudut persegi merupakan garis singgung lingkaran dalam segiempat. Akan ditunjukkan bahwa hasil kali panjang sisi-sisi yang berhadapan sama.

`AB=norm(A-B) //panjang sisi AB`

4

`CD=norm(C-D) //panjang sisi CD`

4

`AD=norm(A-D) // panjang sisi AD`

4

`BC=norm(B-C) //panjang sisi BC`

4

`AB.CD`

16

`AB.BC`

16

Terbukti bahwa hasil kali panjang sisi-sisi yang berhadapan sama yaitu 16. Jadi dapat dipastikan bahwa segiempat tersebut merupakan segiempat garis singgung.

2. Gambarkanlah suatu parabola yang melalui 3 titik yang diketahui.

Petunjuk:

- Misalkan persamaan parabolanya  $y = ax^2 + bx + c$ .

- Substitusikan koordinat titik-titik yang diketahui ke persamaan tersebut.

- Selesaikan SPL yang terbentuk untuk mendapatkan nilai-nilai a, b, c.

`setPlotRange(5); P=[1,0]; Q=[3,0]; R=[0,-3]; ... plotPoint(P,"P"); plotPoint(Q,"Q"); plotPoint(R,"R");`

!images/23030630036\_Marcelline

`sol = solve([a+b=-c,16a+4b=-c,c=-4],[a,b,c])`

`[[a = - 1, b = 5, c = - 4]]`

`function y=x^2 + 4x - 5`

`2 x + 4 x - 5`

`plot2d("-x^2 + 4x - 5", -4, 4, -4, 4) :`

!images/23030630036\_Marcelline

4. Gambarkanlah 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).

*remvalue;*

`P=[-1,-1]; Q=[1,-1]; ... function d1(x,y):=sqrt((x-P[1])^2 + (y - P[2])^2)`

`Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])^2 + (y - P[2])^2) + sqrt((x-Q[1])^2 + (y - Q[2])^2);`

```

fcontour("d2",xmin=-2,xmax=2,ymin=-4,ymax=4,hue=1):
!images/23030630036_Marcelline
reset; plot3d("d2",xmin=-2,xmax=2,ymin=-4,ymax=4):
!images/23030630036_Marcelline
plot2d("abs(x+1)+abs(x-1)",xmin=-2,xmax=2):
!images/23030630036_Marcelline
P= [-1,0];
Q= [1,0];
distance(P,Q)
2
t=linspace(0,2pi,1000); plot2d(1distance(P,Q)cos(t),sin(t),r=3); plot2d(-1,0, add, points);
plot2d(1,0, add, points):
!images/23030630036_Marcelline
—

```

5. Gambarkanlah suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).

```

P=[-2,0]; ... Q=[2,0]; ... t=linspace(-5,5,1000); plot2d(cosh(t),sinh(t),r=3);
plot2d(-cosh(t),-sinh(t), add);... plot2d(-2,0, add, points); plot2d(2,0, add, points):
!images/23030630036_Marcelline

```

## 7 Statistika

23030630036\_MarcellineCalyaPadmarini<sub>EMT</sub>StatistikaNama : MarcellineCalyaPadmarini

Kelas: Matematika E

NIM : 23030630036

EMT untuk Statistika

Pada catatan ini, kita mendemostrasikan plot statistik utama, pengujian, dan pendistribusian dalam Euler.

Mari kita mulai dengan beberapa statistik deskriptif. Ini bukan pengantar statistika. Sehingga anda mungkin membutuhkan beberapa latar belakang untuk mengerti secara mendalam.

Asumsikan pengukuran berikut. Kami ingin menghitung nilai rata-rata dan standar deviasi yang diukur.mean value and the measured standard deviation.

```

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 kotak dan plot whisker untuk data. Pada kasus ini, tidak ada outlier.

```

aspect(1.75); boxplot(M):

```

```

!images/23030630036_Marcelline

```

Kita hitung probabilitas yang menunjukkan nilai lebih besar dari 1005, asumsikan nilai terukur dan distribusi normal.

Semua fungsi dari distribusi dalam Euler diakhiri dengan ...dis dan menghitung distribusi probabilitas kumulatif (CPF).

$$\text{normaldis}(x, m, d) = \int_{-\infty}^x \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t-m}{d})^2} dt.$$

Kita cetak hasil dalam print.

```
print((1-normaldis(1005,mean(M),dev(M)))100,2,unit="
3.07
```

Untuk contoh berikutnya, kita asumsikan jumlah pria berikut dalam rentang ukuran yang diberikan.

```
r=155.5:4:187.5; v=[22,71,136,169,139,71,32,8];
```

Ini adalah plot distribusi.

```
plot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style="
```

```
/"):
```

```
![images/23030630036_Marcelline
```

Kita dapat memasukan data mentah ke dalam tabel.

Tabel adalah metode untuk menyimpan data statistik. Tabel kami harus diberi tiga kolom: Awal jangkauan, akhir jangkauan, dan 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 membutuhkan nilai rata-rata dan statistik lain dari ukuran, kita butuh untuk menghitung nilai tengah dari jangkauan. Kita dapat menggunakan dua kolom pertama dari tabel kita untuk ini.

Simbol "—" berguna untuk memisahkan kolom, fungsi "writetable" digunakan untuk menulis tabel, dengan opsi "labc" digunakan untuk menentukan header kolom.

```
(T[,1]+T[,2])/2 // the midpoint of each interval
```

```
157.5 161.5 165.5 169.5 173.5 177.5 181.5 185.5
```

Namun ini mudah, untuk menentukan rentang dengan vektor [1/2, 1/2]

```
M=fold(r,[0.5,0.5])
```

```
[157.5, 161.5, 165.5, 169.5, 173.5, 177.5, 181.5, 185.5]
```

Sekarang kita dapat menghitung rata-rata dan deviasi dari sampel dengan frekuensi yang ditentukan.

```
m,d=meandev(M,v); m, d,
```

```
169.901234568 5.98912964449
```

Mari kiat tambahkan distribusi nilai dari nilai-nilai ke plot batang diatas.

Rumus untuk distribusi normal dengan rata-rata m dan standar deviasi d adalah:

$$y = \frac{1}{d\sqrt{2\pi}} e^{-\frac{(x-m)^2}{2d^2}}.$$

Karena nilai berada diantara 0 dan 1, untuk membuat plot pada plot batang harus dikalikan sebanyak 4 kali jumlah total data.

```
plot2d("qnormal(x,m,d)sum(v)4", ... xmin=min(r),xmax=max(r),thickness=3,add=1):
```

!images/23030630036\_Marcelline

Tabel

Dalam kamus dari catatan ini, Anda dapat menemukan sebuah file dengan tabel. Data tersebut mewakili hasil survey. Ini adalah empat baris pertama dari file tersebut. Data 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 tersebut terdiri dari 7 kolom angka atau token (kalimat). Kami ingin membaca tabel dari file. Pertama, kami menggunakan terjemahan kami sendiri untuk token.

Untuk ini, kami mendefinisikan himpunan token. Fungsi `strtokens()` mendapatkan vektor string token dari string yang diberikan.

```
mf=["m","f"]; yn=["y","n"]; ev=strtokens("g vg m b vb");
```

Sekarang kita membaca tabel dengan terjemahan ini.

Argumen `tok2`, `tok4`, `dsb` merupakan terjemahan kolom tabel. Argumen ini tidak ada dalam daftar `readtable()`, sehingga anda harus menyediakannya dengan `":="`.

```
MT,hd=readtable("table.dat",tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);  
load over statistics;
```

Untuk mencetak, kita butuh untuk menentukan set token yang sama. Kami mencetak empat baris pertama saja.

```
writetable(MT[1:10],labc=hd,wc=5,tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
```

```
Person Sex Age Titanic Evaluation Tip Problem 1 m 30 n . 1.8 n 2 f 23 y g  
1.8 n 3 f 26 y g 1.8 y 4 m 33 n . 2.8 n 5 m 37 n . 1.8 n 6 m 28 y g 2.8 y 7 f 31  
y vg 2.8 n 8 m 23 n . 0.8 n 9 f 24 y vg 1.8 y 10 m 26 n . 1.8 n
```

Titik "." merepresentasikan sebuah nilai, yang mana tidak tersedia.

Jika kita tidak ingin menentukan token untuk terjemahan 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);
```

Fungsi `readtable()` sekarang mengembalikan suatu set token.

```
tok
```

```
m n f y g vg
```

Tabel berisi entri dari file dengan token yang diterjemahkan menjadi angka.

String khusus `NA=""` ditafsirkan sebagai "Not Available", dan mendapatkan `NAN` (Not A Number) dalam tabel. Terjemahan ini dapat diubah dengan parameter `NA`, dan `NAl`.

```
MT[1]
```

```
[1, 1, 30, 2, NAN, 1.8, 2]
```

Tabel ini berisi angka-angka 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 memasukan output `readtable()` ke dalam daftar.

```
Table=readtable("table.dat",ctok=ctok);
```

Menggunakan kolom token yang sama dan token yang dibaca dari file, kita dapat mencetak tabel. Kita dapat menentukan `ctok`, `tok`, dll. Atau menggunakan daftar tabel.

```
writetable(Table,ctok=ctok,wc=5);
```

```
Person Sex Age Titanic Evaluation Tip Problem 1 m 30 n . 1.8 n 2 f 23 y g
1.8 n 3 f 26 y g 1.8 y 4 m 33 n . 2.8 n 5 m 37 n . 1.8 n 6 m 28 y g 2.8 y 7 f 31
y vg 2.8 n 8 m 23 n . 0.8 n 9 f 24 y vg 1.8 y 10 m 26 n . 1.8 n 11 f 23 y vg 1.8
y 12 m 32 y g 1.8 n 13 m 29 y vg 1.8 y 14 f 25 y g 1.8 y 15 f 31 y g 0.8 n 16 m
26 y g 2.8 n 17 m 37 n . 3.8 n 18 m 38 y g . n 19 f 29 n . 3.8 n 20 f 28 y vg 1.8
n 21 f 28 y m 2.8 y 22 f 28 y vg 1.8 y 23 f 38 y g 2.8 n 24 f 27 y m 1.8 y 25 m
27 n . 2.8 y
```

Fungsi `tablecol()` mengembalikan nilai kolom dari tabel, melewati setiap baris dengan nilai `NAN("."` dalam file), dan indeks kolom, yang berisi nilai-nilai ini.

```
c,i=tablecol(MT,[5,6]);
```

Kita dapat menggunakan ini untuk mengekstrak kolom dari tabel untuk tabel baru.

```
j=[1,5,6]; writetable(MT[i,j],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 untuk mengekstrak tabel itu sendiri dari daftar Tabel dalam kasus ini.

```
MT=Table[1];
```

Tentu saja, kita juga dapat menggunakannya untuk menentukan nilai rata-rata kolom atau nilai statistik lainnya.

```
mean(tablecol(MT,6))
```

```
2.175
```

Fungsi `getstatistics()` mengembalikan elemen dalam vektor dan jumlahnya. Kami menerapkannya pada nilai "m" dan "f" di kolom kedua kami.

```
xu,count=getstatistics(tablecol(MT,2)); xu, count,
```

```
[1, 3] [12, 13]
```

Kita dapat mencetak hasil dengan tabel baru.

```
writetable(count',labr=tok[xu])
```

```
m 12 f 13
```

Fungsi `selecttable()` mengembalikan tabel baru dengan nilai dalam satu kolom yang dipilih dari indeks vektor. Pertama, kita cari indeks dari dua nilai kita dalam tabel token.

```
v:=indexof(tok,["g","vg"])
```

```
[5, 6]
```

Sekarang kita dapat memilih baris tabel, yang mana memiliki salah satu nilai dalam v pada baris ke-5.

```
MT1:=MT[selectrows(MT,5,v)]; i:=sortedrows(MT1,5);
```

Kemudian kita dapat mencetak tabel, dengan nilai yang diekstrak dan diurutkan pada kolom ke-5.

```
writetable(MT1[i],labc=hd,ctok=ctok,tok=tok,wc=7);
```

```
Person Sex Age Titanic Evaluation Tip Problem 2 f 23 y g 1.8 n 3 f 26 y g
1.8 y 6 m 28 y g 2.8 y 18 m 38 y g . n 16 m 26 y g 2.8 n 15 f 31 y g 0.8 n 12 m
32 y g 1.8 n 23 f 38 y g 2.8 n 14 f 25 y g 1.8 y 9 f 24 y vg 1.8 y 7 f 31 y vg 2.8
n 20 f 28 y vg 1.8 n 22 f 28 y vg 1.8 y 13 m 29 y vg 1.8 y 11 f 23 y vg 1.8 y
```

Untuk statistik berikutnya, kami ingin menghubungkan dua kolom tabel. Jadi kami mengekstrak kolom 2 dan serta mengurutkan tabel.

```
i:=sortedrows(MT,[2,4]); ... writetable(tablecol(MT[i],[2,4]),ctok=[1,2],tok=tok)
m n m n m n m n m n m n m 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
```

Dengan `getstatistics()`, kita juga dapat menghubungkan hitungan dalam dua kolom tabel satu sama lain.

```
MT24=tablecol(MT,[2,4]); ... xu1,xu2,count=getstatistics(MT24[1],MT24[2]);
... writetable(count,labr=tok[xu1],labc=tok[xu2])
```

```
n y m 7 5 f 1 12
```

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.

```
MT2,hd,tok2,hdr=readtable(filename, clabs, rlabs); ... writetable(MT2,labr=hdr,labc=hd)
```

```
n y m 7 5 f 1 12
```

dan menghapus file tersebut.

```
fileremove(filename);
```

Distribusi

Dengan `plot2d`, ada metode yang lebih mudah untuk membuat plot distribusi data eksperimen.

```
p=normal(1,1000); //1000 random normal-distributed sample p
plot2d(p,distribution=20,style="
/"); // plot the random sample p
plot2d("qnormal(x,0,1)",add=1): // add the standard normal distribution
plot
```

```
![images/23030630036_Marcelline
```

Harap catat perbedaan antara plot batang (sampel) dan kurva normal (distribusi nyata). Masukkan kembali tiga perintah untuk melihat hasil pengambilan sampel lainnya.

Berikut adalah perbandingan 10 simulasi dari 1000 nilai yang terdistribusi menggunakan plot kotak. Plot ini menunjukkan median, kuartil 25

```
p=normal(10,1000); boxplot(p):
```

```
![images/23030630036_Marcelline
```

Untuk menghasilkan bilangan bulat acak. Euler memiliki `inrandom`. Mari kita simulasikan lemparan dadu dan plot distribusinya.

Kita gunakan fungsi `getmultiplicities(v,x)`, untuk menghitung seberapa sering elemen `v` muncul di `x`. Kemudian kita plot hasilnya dengan menggunakan `columnplot()`.

```
k=intrandom(1,6000,6); ... columnplot(getmultiplicities(1:6,k)); ...
ygrid(1000,color=red):
```

```
![images/23030630036_Marcelline
```

Sementara `intrandom(n,m,k)` mengembalikan bilangan bulat terdistribusi seragam dari 1 ke `k`, dimungkinkan untuk menggunakan distribusi bilangan bulat lainnya dengan `randpint()`.

Pada contoh berikut, probabilitas untuk 1,2,3 secara berurutan 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 random dari beberapa distribusi. Coba lihat referensinya.

Misalkan, kita coba untuk membuat distribusi eksponensial. Sebuah variabel acak kontinu  $X$  dikatakan memiliki distribusi eksponensial, jika PDF-nya diberikan oleh

$$f_X(x) = \lambda e^{-\lambda x}, \quad x > 0, \quad \lambda > 0,$$

dengan parameter

$$\lambda = \frac{1}{\mu}, \quad \mu \text{ adalah mean, dan } \text{distribusi } X \sim \text{Exponential}(\lambda).$$

```
plot2d(randexponential(1,1000,2), distribution):
```

```
![images/23030630036_Marcelline
```

Untuk beberapa distribusi, Euler dapat menghitung fungsi distribusi dan inversnya.

```
plot2d("normaldis",-4,4):
```

```
![images/23030630036_Marcelline
```

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/23030630036_Marcelline
```

$$\text{normaldis}(x, m, d) = \int_{-\infty}^x \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-m}{d}\right)^2} dt.$$

Probabilitas berada di area hijau adalah sebagai berikut.

```
normaldis(5,1,1.5)-normaldis(2,1,1.5)
```

```
0.248662156979
```

Ini dapat dihitung secara numerik dengan menggunakan integral berikut.

$$\int_2^5 \frac{1}{1.5\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-1}{1.5}\right)^2} dx.$$

```
gauss("qnormal(x,1,1.5)",2,5)
```

0.248662156979

Mari kita bandingkan distribusi binomial dengan distribusi normal dengan mean yang sama serta deviasi yang sama. Fungsi `invbindis()` menyelesaikan interpolasi linier antara nilai bilangan bulat.

```
invbindis(0.95,1000,0.5), invnormaldis(0.95,500,0.5*sqrt(1000))  
525.516721219 526.007419394
```

Fungsi `qdis()` adalah densitas dari distribusi chi-square. Seperti biasa, Euler memetakan vektor ke fungsi ini. Dengan demikian, kita mendapatkan plot dari semua distribusi chi-square dengan derajat 5 sampai 30 dengan mudah dengan cara berikut.

```
plot2d("qchidis(x,(5:5:50))",0,50):
```

```
!images/23030630036_Marcelline
```

Euler memiliki fungsi akurat untuk mengevaluasi distribusi. Mari kita periksa `chidis()` dengan sebuah integral.

Penamaan mencoba untuk konsisten. Misalnya,

- distribusi chi-square adalah `chidis()`,
- fungsi invers adalah `invchidis()`,
- fungsi kepadatan adalah `qchidis()`.

Komplemen dari distribusi (ekor atas) adalah `chicdis()`.

```
chidis(1.5,2), integrate("qchidis(x,2)",0,1.5)  
0.527633447259 0.527633447259
```

Distribusi Diskrit

Untuk mendefinisikan 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]
```

Artinya dengan probabilitas `wd[i+1]-wd[i]` kita menghasilkan nilai acak `i`.

Ini adalah distribusi yang hampir seragam. Mari kita mendefinisikan generator nomor acak untuk ini. Fungsi `find(v,x)` menemukan nilai `x` dalam 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 sangat banyak iterasi.

```
columnplot(getmultiplicities(1:6,wrongdice(1,1000000))):
```

```
!images/23030630036_Marcelline
```

Ini adalah fungsi sederhana untuk memeriksa distribusi seragam dengan nilai `1...K` dalam `v`. Kita menerima hasilnya, jika untuk semua frekuensi

$$\left| f_i - \frac{1}{K} \right| < \frac{\delta}{\sqrt{n}}.$$

```
function checkrandom (v, delta=1) ...
```

```
  K=max(v); n=cols(v); fr=getfrequencies(v,1:K); return max(fr/n-1/K)_delta/sqrt(n);  
endfunction i/prej Memang fungsi tersebut menolak distribusi seragam.
```

```
  checkrandom(wrongdice(1,1000000))
```

```

0
Dan ia menerima generator acak bawaan.
checkrandom(intrandom(1,1000000,6))
1
Kita dapat menghitung distribusi binomial. Pertama ada binomialsu(m),
yang mengembalikan probabilitas i atau kurang hit dari percobaan.
bindis(410,1000,0.4)
0.751401349654
Fungsi Beta terbalik digunakan untuk menghitung interval kepercayaan Clopper-
Pearson untuk paramter p. Tingkat default adalah alpha.
Arti dari interval ini adalah jika p diluar interval, maka hasil pengamatan
410 dalam 1000 jarang terjadi.
clopperpearson(410,1000)
[0.37932, 0.441212]
Perintah berikut adalah cara langsung untuk mendapatkan hasil di atas.
Tetapi untuk n besar, penjumlahan langsung 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, invbinsu(m) menghitung invers dari binomialsu(m).
invbindis(0.75,1000,0.4)
409.932733047
Di Bridge, kami mengasumsikan 5 kartu yang 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).
2hypergeomsu(m(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
Plotting Data
Untuk plot data, kami mencoba hasil pemilu Jerman sejak tahun 1990,
diukur dalam kursi.
BW := [ ... 1990,662,319,239,79,8,17; ... 1994,672,294,252,47,49,30; ...
1998,669,245,298,43,47,36; ... 2002,603,248,251,47,55,2; ... 2005,614,226,222,61,51,54;
... 2009,622,239,146,93,68,76; ... 2013,631,311,193,0,63,64];
Untuk pesta, kami menggunakan serangkaian nama.
P:=[”CDU/CSU”, ”SPD”, ”FDP”, ”Gr”, ”Li”];
Mari kita mencetak persentase dengan baik.
Pertama kita ekstrak kolom yang diperlukan. Kolom 3 sampai 7 adalah kursi
masing-masing partai, dan kolom 2 adalah jumlah kursi. kolom adalah tahun
pemilihan.
BT:=BW[,3:7]; BT:=BT/sum(BT); YT:=BW[,1]’;
Kemudian kami mencetak statistik dalam bentuk tabel. Kami menggunakan
nama sebagai tajuk kolom, dan tahun sebagai tajuk untuk baris. Lebar default

```

untuk kolom adalah `wc=10`, tetapi kami lebih memilih output 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 mengekstrak jumlah persentase dua partai besar yang menunjukkan bahwa partai-partai kecil telah memperoleh rekaman di parlemen hingga 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/23030630036_Marcelline
Tentukan beberapa warna untuk setiap pesta.
CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.8,0,0)];
```

Sekarang kita dapat memplot hasil pemilu 2009 dan perubahannya menjadi satu plot 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/23030630036_Marcelline
```

Plot data menggabungkan deretan data statistik dalam satu plot.

```
J:=BW[,1]'; DP:=BW[,3:7]'; ... dataplot(YT,BT',color=CP); ... label-
box(P,colors=CP,styles="[]", points,w=0.2,x=0.3,y=0.4):
!images/23030630036_Marcelline
```

Plot kolom 3D menunjukkan baris data statistik dalam bentuk kolom. Kami menyediakan label untuk baris dan kolom. sudut adalah sudut pandang.

```
columnsplot3d(BT,scols=P,srows=YT, ... angle=30°,ccols=CP):
!images/23030630036_Marcelline
```

Representasi lain adalah plot mosaik. Perhatikan bahwa kolom plot mewakili kolom matriks di sini. Karena panjangnya label CDU/CSU, kami mengambil jendela yang lebih kecil dari biasanya.

```
shrinkwindow( smaller); ... mosaicplot(BT',srows=YT,scols=P,color=CP,style="");
... shrinkwindow():
!images/23030630036_Marcelline
```

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/23030630036_Marcelline
```

Berikut adalah jenis plot lainnya

```
starplot(normal(1,10)+4,lab=1:10, rays):
!images/23030630036_Marcelline
```

Beberapa plot di `plot2d` bagus untuk statika. Berikut adalah plot impuls dari data acak, terdistribusi secara merata di  $[0,1]$ .

```

plot2d(makeimpulse(1:10,random(1,10)), bar):
!images/23030630036_Marcelline

```

Tetapi untuk data yang terdistribusi secara eksponensial, kita mungkin memerlukan plot logaritmik.

```

logimpulseplot(1:10,-log(random(1,10))10):
!images/23030630036_Marcelline

```

Fungsi `columnplot()` lebih mudah digunakan, karena hanya membutuhkan vektor nilai. Selain itu, ia dapat mengatur labelnya ke apa pun yang kita inginkan, kita sudah mendemonstrasikannya dalam tutorial ini.

Ini adalah aplikasi lain, di mana kita menghitung karakter dalam sebuah kalimat dan menyusun 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/23030630036_Marcelline

```

Dimungkinkan juga untuk mengatur sumbu secara manual.

```

n=10; p=0.4; i=0:n; x=bin(n,i)pi(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/23030630036_Marcelline

```

Berikut ini adalah cara untuk memplot frekuensi bilangan dalam sebuah vektor.

Kami membuat vektor bilangan bulat bilangan acak 1 hingga 6.

```

v=intrandom(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/23030630036_Marcelline

```

Kami ingin menunjukkan fungsi untuk distribusi nilai empiris.

```

x=normal(1,20);

```

Fungsi `empdist(x,vs)` membutuhkan array nilai yang diurutkan. Jadi kita harus mengurutkan `x` sebelum kita dapat menggunakannya.

```

xs=sort(x);

```

Kemudian kami memplot distribusi empiris dan beberapa batang kepadatan menjadi satu plot. Alih-alih plot batang untuk distribusi, kami menggunakan plot gigi gergaji kali ini.

```

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/23030630036_Marcelline

```

Plot pencar 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/23030630036\_Marcelline

Seringkali, kita ingin membandingkan dua sampel dari distribusi yang berbeda. Ini dapat dilakukan dengan plot kuantil-kuantil.

Untuk pengujian, kami mencoba distribusi student-t dan distribusi eksponensial.

```
x=randt(1,1000,5); y=randnormal(1,1000,mean(x),dev(x)); ... plot2d("x",r=6,style="--",yl="normal",xl="student-t", vertical); ... plot2d(sort(x),sort(y), points,color=red,style="x", add):
```

!images/23030630036\_Marcelline

Plot dengan jelas menunjukkan bahwa nilai terdistribusi normal cenderung lebih kecil di ujung ekstrim.

Jika kita memiliki dua distribusi dengan ukuran yang berbeda, kita dapat memperluas yang lebih kecil atau mengecilkan yang lebih besar. Fungsi berikut 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/23030630036\_Marcelline

Regresi dan Korelasi

Regresi linier dapat dilakukan dengan fungsi polyfit() atau berbagai fungsi fit.

Sebagai permulaan, kami menemukan garis regresi untuk data univariat dengan polifit(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 non-weighted dan weighted fit. Pertama, koefisien kecocokan linier.

```
p=polyfit(x,y,1)
[0.733333, 0.812121]
```

Sekarang koefisien dengan bobot yang menekankan nilai terakhir.

```
w = "exp(-(x-10)2/10)"; pw = polyfit(x,y,1,w=w(x))
[4.71566, 0.38319]
```

kita memasukkan semuanya ke dalam satu plot untuk titik dan garis regresi, dan untuk bobot yang digunakan.

Sebagai permulaan, kami menemukan garis regresi untuk data univariat dengan polifit(x,y,1).

```
figure(2,1); ... figure(1); statplot(x,y,"b",xl="Regression"); ... plot2d("evalpoly(x,p)", add,color=blue,style="o"); ... plot2d("evalpoly(x,pw)",5,10, add,color=red,style="--"); ... figure(2); plot2d(w,1,10, filled,style="r",fillcolor=red,xl=w); ... figure(0):
```

!images/23030630036\_Marcelline

Sebagai 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 daripada membiarkan readtable() mengumpulkan terjemahan.



```

MS,hd:=readtable("table1.dat",tok2:=["m","f"]); ... writetable(MS,labc=hd,tok2:=["m","f"]);
Person Sex Age Mother Father Siblings 1 m 29 58 61 1 2 f 26 53 54 2 3 m
24 49 55 1 4 f 25 56 63 3 5 f 25 49 53 0 6 f 23 55 55 2 7 m 23 48 54 2 8 m 27 56
58 1 9 m 25 57 59 1 10 m 24 50 54 1 11 f 26 61 65 1 12 m 24 50 52 1 13 m 29
54 56 1 14 m 28 48 51 2 15 f 23 52 52 1 16 m 24 45 57 1 17 f 24 59 63 0 18 f 23
52 55 1 19 m 24 54 61 2 20 f 23 54 55 1

```

Bagaimana usia bergantung satu sama lain? Kesan pertama datang dari scatterplot berpasangan.

```
scatterplots(tablecol(MS,3:5),hd[3:5]):
```

!images/23030630036\_Marcelline

Jelas bahwa usia ayah dan ibu 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 model yang salah. Garis regresinya adalah  $s=17+0,74t$ , di mana  $t$  adalah usia ibu dan  $s$  usia ayah. Perbedaan usia mungkin sedikit bergantung pada usia, tetapi tidak terlalu banyak.

Sebaliknya, kami menduga 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 pencar.

```
plot2d(cs[1],cs[2], points); ... plot2d("evalpoly(x,ps)",color=red,style=".", add);
```

```
... plot2d("x+da",color=blue, add):
```

!images/23030630036\_Marcelline

Berikut adalah plot kotak dari dua zaman. Ini hanya menunjukkan, bahwa usianya berbeda.

```
boxplot(cs,["mothers","fathers"]):
```

!images/23030630036\_Marcelline

Sangat menarik bahwa perbedaan median tidak sebesar perbedaan rata-rata.

```
median(cs[2])-median(cs[1])
```

```
1.5
```

Koefisien korelasi menunjukkan korelasi positif.

```
correl(cs[1],cs[2])
```

```
0.7588307236
```

Korelasi peringkat adalah ukuran untuk urutan yang sama di kedua vektor.

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. Sebagai contoh, kita definsikan fungsi skewness.

$$sk(x) = \sqrt{n} \sum_i (x_i - m)^3 \left( \sum_i (x_i - m)^2 \right)^{3/2}$$

dengan m adalah mean dari x.

```
function skew (x:vector) ...
m=mean(x); return sqrt(cols(x))*sum((x-m)^3)/(sum((x-m)^2))^(3/2); endfunction <


```

/pret > SepertiyangAndalihat, kitadapatdenganmudahmenggunakanbahasamatriksuntukmendapatkanimpl
data=normal(20); skew(normal(10))
-0.485603444202
```


```

Berikut adalah fungsi lain, disebut sebagai koefisien Pearson skewness.

```
function skew1 (x) := 3(mean(x)-median(x))/dev(x)
skew1(data)
0.268537236365
```

Simulasi Monte Carlo

Euler dapat digunakan untuk mensimulasikan kejadian acak. Kita telah melihat contoh sederhana diatas. Ini adalah satu lagi, yang mensimulasikan 1000 kali 3 lemparan dadu, dan meminta distribusi jumlah.

```
ds:=sum(intrandom(1000,3,6)); fs=getmultiplicities(3:18,ds)
[5, 16, 19, 50, 75, 97, 120, 139, 127, 119, 92, 66, 40, 23, 10, 2]
Kita dapat memplot ini sekarang
columnplot(fs,lab=3:18):
```

!images/23030630036\_Marcelline

Untuk menentukan distribusi yang diharapkan tidak begitu mudah. Kami menggunakan rekursi lanjutan untuk ini.

Fungsi berikut menghitung banyaknya cara bilangan k dapat direpresentasikan sebagai jumlah n bilangan dalam rentang 1 sampai m. Ia bekerja secara rekursif dengan cara yang jelas.

```
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 ;/pret
countways(5:25,5,5)
[1, 5, 15, 35, 70, 121, 185, 255, 320, 365, 381, 365, 320, 255, 185, 121, 70, 35,
15, 5, 1]
```

```
cw=countways(3:18,3,6)
[1, 3, 6, 10, 15, 21, 25, 27, 27, 25, 21, 15, 10, 6, 3, 1]
```

Kita tambahkan nilai harapan ke dalam plot.

```
plot2d(cw/6^31000, add); plot2d(cw/6^31000, points, add) :
```

!images/23030630036\_Marcelline

Untuk simulasi lain, simpangan nilai rata-rata dari n 0-1-variabel acak terdistribusi normal adalah  $1/\sqrt{n}$

```
longformat; 1/sqrt(10)
0.316227766017
```

Mari kita periksa ini dengan simulasi. Kami memproduksi 10000 kali 10 vektor acak.

```
M=normal(10000,10); dev(mean(M)')
0.316948166828
plot2d(mean(M)', distribution):
!images/23030630036_Marcelline
```

Nilai tengah dari 10 0-1-distribusi normal bialngan acak memiliki simpangan yang lebih besar.

```
dev(median(M'))
0.373307244349
```

Since we can easily generate random walks, we can simulate the Wiener process. We take 1000 steps of 1000 processes. We then plot the standard deviation and the mean of the n-th step of these processes together with the expected values in red.

```
reset; 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/23030630036_Marcelline
```

Tes

Tes adalah alat penting dalam statistik. Dalam Euler, banyak tes yang diimplementasikan. Semua tes tersebut mengembalikan kesalahan yang kami terima, jika kami menolak hipotesis nol.

Sebagai contoh, kita akan menguji lemparan dadu untuk distribusi seragam. Pada 600 lemparan, kita mendapat nilai berikut, yang dimasukan dalam uji chi-square.

```
chitest([90,103,114,101,103,89],dup(100,6)')
0.498830517952
```

Uji chi-square juga memiliki mode, yang menggunakan simulasi Monte Carlo untuk menguji statistik. Hasilnya harus hampir sama. Parameter gt;p menginterpretasikan vektor-y sebagai vektor probabilitas.

```
chitest([90,103,114,101,103,89],dup(1/6,6)', p, montecarlo)
0.506
```

Kesalahan ini terlalu besar. Sehingga kita tidak dapat menolak distribusi normal. Ini tidak membuktikan bahwa dadu kita adil. Namun kami tidak dapat menolak hipotesis kami.

Kemudian kita hasilkan 1000 lemparan dadu dengan menggunakan generator angka acak, dan lakukan tes yang sama.

```
n=1000; t=random([1,n6]); chitest(count(t6,6),dup(n,6)')
0.39595925664
```

Mari kita uji untuk nilai rata-rata (mean) 100 dengan uji-t.

```
s=200+normal([1,100])10; ... ttest(mean(s),dev(s),100,200)
0.498440735333
```

Fungsi ttest() membutuhkan nilai rata-rata, deviasi (simpangan), jumlah data, dan nilai rata-rata yang akan diuji.

Sekarang mari kita periksa dua pengukuran dengan rata-rata yang sama. Kita menolak hipotesis apabila mereka memiliki rata-rata yang sama, jika hasilnya lt;0.05

```
tcomparedata(normal(1,10),normal(1,10))
0.444876485838
```

Jika kita menambah sebuah bias ke dalam satu distribusi, kita akan mendapatkan lebih banyak penolakan. Ulangi simulasi ini beberapa kali untuk melihat efeknya.

```
tcomparedata(normal(1,10),normal(1,10)+2)
8.36733514392e-05
```

Pada contoh berikutnya, kita menghasilkan 20 lemparan dadu acak sebanyak 100 kali dan menghitung yang ada didalamnya. Harus ada  $20/6=3.3$  yang rata-rata.

```
R=random(100,20); R=sum(R6i=1)'; mean(R)
3.13
```

Sekarang kita bandingkan jumlah satu dengan distribusi binomial. Pertama kita plot distribusinya.

```
plot2d(R,distribution=max(R)+1,even=1,style="
/"):
![images/23030630036_Marcelline
t=count(R,21);
Kemudian kita hitung nilai harapan.
n=0:20; b=bin(20,n)(1/6)^n(5/6)^(20-n)100;
Kita harus mengumpulkan beberapa angka untuk mendapatkan kategori
yang cukup besar.
t1=sum(t[1:2])—t[3:7]—sum(t[8:21]); ... b1=sum(b[1:2])—b[3:7]—sum(b[8:21]);
Uji chi-square menolak hipotesis apabila distribusi kita adalah distribusi
binomial, jika hasilnya lt;0.05.
chitest(t1,b1)
0.302019360722
```

Pada contoh berikut, berisi hasil dua kelompok orang (pria dan wanita, katakanlah) memberikan suara untuk satu dari enam partai.

```
A=[23,37,43,52,64,74;27,39,41,49,63,76]; ... writetable(A,wc=6,labr=["m","f"],labc=1:6)
1 2 3 4 5 6 m 23 37 43 52 64 74 f 27 39 41 49 63 76
```

Kami ingin menguji independensi suara dari jenis kelamin. Tabel uji chi-square melakukan ini. Hasilnya terlalu besar untuk menolak independensi. Jadi kami tidak dapat mengatakan jika voting bergantung pada jenis kelamin dari data ini.

```
tabletest(A)
0.990701632326
```

Berikut ini adalah tabel harapan, kita asumsikan frekuensi pemungutan suara yang diamati.

```
writetable(expectedtable(A),wc=6,dc=1,labr=["m","f"],labc=1:6)
1 2 3 4 5 6 m 24.9 37.9 41.9 50.3 63.3 74.7 f 25.1 38.1 42.1 50.7 63.7 75.3
```

Kita dapat menghitung koefisien kontingensi yang dikoreksi. Karena sangat dekat dengan 0, kami menyimpulkan bahwa pemungutan suara tidak tergantung pada jenis kelamin.

```
contingency(A)
0.0427225484717
```

Beberapa tes lagi

Selanjutnya kita gunakan analisis varian (Uji-F) untuk menguji tiga sampel data dari distribusi normal untuk nilai rata-rata yang sama. Metode ini disebut ANOVA (analysis of variance). Dalam Euler, fungsi dari varanalysis() digunakan.

```

x1=[109,111,98,119,91,118,109,99,115,109,94]; mean(x1),
106.545454545
x2=[120,124,115,139,114,110,113,120,117]; mean(x2),
119.111111111
x3=[120,112,115,110,105,134,105,130,121,111]; mean(x3)
116.3
varanalysis(x1,x2,x3)
0.0138048221371

```

Ini berarti, kita menolak hipotesis dengan nilai rata-rata yang sama. Kita melakukan ini dengan sebuah kesalahan probabilitas 1.3

Ada juga uji median (nilai tengah) yang menolak sampel data dengan distribusi rata-rata berbeda menguji median sampel bersatu.

```

a=[56,66,68,49,61,53,45,58,54];
b=[72,81,51,73,69,78,59,67,65,71,68,71];
mediantest(a,b)
0.0241724220052

```

Tes lain tentang kesetaraan adalah tes peringkat. Ini jauh lebih tajam daripada tes median.

```

ranktest(a,b)
0.00199969612469

```

Dalam contoh berikut, kedua distribusi memiliki mean yang sama.

```

ranktest(random(1,100),random(1,50)-1)
0.30631773688

```

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, jika a lebih baik dari b.

```

signtest(a,b)
0.0546875

```

Ini terlalu banyak kesalahan. Kita tidak dapat menolak bahwa a sama baiknya dengan b.

Tes Wilcoxon lebih tajam dari tes ini, tetapi bergantung pada nilai kuantitatif perbedaan.

```

wilcoxon(a,b)
0.0296680599405

```

Mari kita coba dua tes lagi menggunakan seri yang dihasilkan.

```

wilcoxon(normal(1,20),normal(1,20)-1)
0.00499819423799
wilcoxon(normal(1,20),normal(1,20))
0.216524144214

```

Angka Acak

Berikut ini adalah pengujian untuk pembangkit bilangan acak. Euler menggunakan generator yang sangat baik, sehingga kita tidak perlu mengharapkan masalah.

Pertama kita hasilkan sepuluh juta angka acak di  $[0,1]$ .

```

n:=10000000; r:=random(1,n);
Kemudian kita hitung jarak antara dua angka kurang dari 0.05.
a:=0.05; d:=differences(nonzeros(rja));
Akhirnya, kita dapat membuat plot beberapa kali, setiap jarak yang terjadi,
dan membandingkan dengan nilai yang diharapkan.
m=getmultiplicities(1:100,d); plot2d(m); ... plot2d("n(1-a)(x-1)a2", color =
red, add) :

```

!images/23030630036\_Marcelline

Bersihkan data.

```
remvalue n;
```

Pengenalan untuk Pengguna dari R Project

Jelas, 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 terbiasa dengan R, tetapi perlu mengetahui perbedaan sintaks EMT dan R. Kami mencoba memberikan gambaran tentang hal-hal yang jelas dan kurang jelas yang perlu Anda ketahui.

Selain itu, kami mencari cara untuk bertukar data antara kedua sistem.

Harap dicatat bahwa ini adalah pekerjaan yang masih dalam tahap pengerjaan.

Sintaks Dasar

Hal pertama yang Anda pelajari dalam R yaitu membuat sebuah vektor. Dalam EMT, perbedaan utama adalah operator : dapat mengambil ukuran langkah. Selain itu, ia memiliki daya ikat yang rendah.

```
n=10; 0:n/20:n-1
```

```
[0, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6, 6.5, 7, 7.5, 8, 8.5, 9]
```

Fungsi c() tidak ada. Ini memungkinkan untuk menggunakan vektor untuk menggabungkan sesuatu.

Pada contoh berikut, seperti banyak contoh lain, dari "Introduction to R" yang disertakan dengan R project. Jika Anda membaca PDF ini, Anda dapat menemukan bahwa saya mengikuti jalannya dalam tutorial ini.

```
x=[10.4, 5.6, 3.1, 6.4, 21.7]; [x,0,x]
```

```
[10.4, 5.6, 3.1, 6.4, 21.7, 0, 10.4, 5.6, 3.1, 6.4, 21.7]
```

Operator titik dua dengan ukuran langkah EMT diganti dengan fungsi seq() di R. Kita bisa menulis fungsi ini di EMT.

```
function seq(a,b,c) := a:b:c; ... seq(0,-0.1,-1)
```

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

Fungsi rep() dari R tidak ada di EMT. Untuk input vektor, dapat ditulis sebagai berikut.

```
function rep(x:vector,n:index) := flatten(dup(x,n)); ... rep(x,2)
```

```
[10.4, 5.6, 3.1, 6.4, 21.7, 10.4, 5.6, 3.1, 6.4, 21.7]
```

Perhatikan bahwa "=" atau ":=" digunakan untuk tugas. Operator "-gt;" digunakan untuk unit di EMT

```
125km - " miles"
```

```
77.6713990297 miles
```

Operator "lt;" untuk penugasan tetap menyesatkan, dan bukan ide yang baik untuk R. Berikut ini akan membandingkan a dan -4 di EMT.

```
a=2; aj-4
```

```
0
```

Di R, "alt;-4lt;3" berfungsi, tetapi "alt;-4lt;-3" tidak. Saya juga memiliki ambiguitas serupa di EMT, tetapi mencoba menghilangkannya perlahan-lahan.

EMT dan R memiliki vektor bertipe boolean. Namun di EMT, angka 0 dan 1 digunakan untuk mewakili salah dan benar. Di R, nilai true dan false dapat digunakan dalam aritmatika biasa seperti di EMT.

```
xj5,
```

```
[0, 0, 1, 0, 0] [0, 0, 3.1, 0, 0]
```

EMT melempar kesalahan atau menghasilkan NAN tergantung pada tanda "kesalahan".

```
errors off; 0/0, isNAN(sqrt(-1)), errors on;
```

```
NAN 1
```

String sama di R dan EMT. Keduanya berada di lokal saat ini, bukan di Unicode.

Di R ada paket untuk Unicode. Di EMT, sebuah 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
```

Berikut ini mungkin atau mungkin tidak ditampilkan dengan benar di sistem Anda sebagai A dengan titik dan garis di atasnya. Itu tergantung pada font yang Anda gunakan.

```
chartoutf([480])
```

Penggabungan string dilakukan dengan "+" atau "—". Ini dapat mencakup angka, yang akan dicetak dalam format saat ini.

```
"pi = "+pi
```

```
pi = 3.14159265359
```

```
Pengindeks-an
```

Sebagian besar waktu, ini akan berfungsi seperti pada R.

Tetapi EMT akan menginterpretasikan indeks dari belakang vektor, sedangkan R menginterpretasikan x[n] sebagai x tanpa elemen ke-n.

```
x, x[1:3], x[-2]
```

```
[10.4, 5.6, 3.1, 6.4, 21.7] [10.4, 5.6, 3.1] 6.4
```

Perilaku R dapat dicapai dalam EMT dengan drop().

```
drop(x,2)
```

```
[10.4, 3.1, 6.4, 21.7]
```

Vektor logis 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]
```

Tetapi nama untuk indeks tidak dimungkinkan di EMT. Untuk paket statistik, ini mungkin sering diperlukan untuk memudahkan akses ke elemen vektor.

Untuk meniru perilaku ini, kita dapat mendefinisikan fungsi sebagai berikut.

```
function sel (v,i,s) := v[indexof(s,i)]; ... s=["first","second","third","fourth"];
sel(x,["first","third"],s)
```

```
Trying to overwrite protected function sel! Error in: function sel (v,i,s) :=
v[indexof(s,i)]; ... ...
```

```
Trying to overwrite protected function sel! Error in: function sel (v,i,s) :=
v[indexof(s,i)]; ... ...
```

```
Trying to overwrite protected function sel! Error in: function sel (v,i,s) :=
v[indexof(s,i)]; ... ...
```

```
Trying to overwrite protected function sel! Error in: function sel (v,i,s) :=
v[indexof(s,i)]; ... ... [10.4, 3.1]
```

Tipe Data

EMT memiliki lebih banyak tipe data tetap daripada R. Jelas, di R ada vektor yang tumbuh. Anda dapat mengatur vektor numerik kosong `v` dan menetapkan nilai ke elemen `v[17]`. Ini tidak mungkin di EMT.

Berikut ini agak tidak efisien.

```
v=[]; for i=1 to 10000; v=v—i; end;
```

EMT sekarang akan membuat vektor dengan `v` dan `i` ditambahkan pada tumpukan dan menyalin vektor itu kembali ke variabel global `v`.

Semakin efisien pra-mendefinisikan vektor.

```
v=zeros(10000); for i=1 to 10000; v[i]=i; end;
```

Untuk mengubah jenis tanggal di EMT, Anda dapat menggunakan fungsi seperti `complex()`.

```
complex(1:4)
[ 1+0i , 2+0i , 3+0i , 4+0i ]
```

Konversi ke string hanya dimungkinkan untuk tipe data dasar. Format saat ini digunakan untuk rangkaian string sederhana. Tetapi 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 |length(v) then s=s+" ";
endif; end; return s+""; endfunction i/pre; tostr(linspace(0,1,10))
[0.00,0.10,0.20,0.30,0.40,0.50,0.60,0.70,0.80,0.90,1.00]
```

Untuk komunikasi dengan Maxima, terdapat fungsi `convertmxm()`, yang juga dapat digunakan untuk memformat vektor untuk output.

```
convertmxm(1:10)
[1,2,3,4,5,6,7,8,9,10]
```

Untuk Latex perintah `tex` dapat digunakan untuk mendapatkan perintah Latex

```
tex([1,2,3])
[1, 2, 3]
```

Faktor dan Tabel

Dalam pengantar R ada contoh dengan apa yang disebut faktor.

Berikut ini adalah daftar wilayah dari 30 negara bagian.



```
austates = ["tas", "sa", "qld", "nsw", "nsw", "nt", "wa", "wa", ... "qld",
"vic", "nsw", "vic", "qld", "qld", "sa", "tas", ... "sa", "nt", "wa", "vic",
"qld", "nsw", "nsw", "wa", ... "sa", "act", "nsw", "vic", "vic", "act"];
```

Asumsikan, kita memiliki pendapatan yang sesuai di setiap negara bagian.

```
incomes = [60, 49, 40, 61, 64, 60, 59, 54, 62, 69, 70, 42, 56, ... 61, 61, 61,
58, 51, 48, 65, 49, 49, 41, 48, 52, 46, ... 59, 46, 58, 43];
```

Sekarang, kami ingin menghitung rata-rata pendapatan di wilayah tersebut. Menjadi program statistik, R memiliki `factor()` dan `tapply()` untuk ini.

EMT dapat melakukannya dengan menemukan indeks wilayah dalam daftar wilayah yang 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 loop 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 > Ini agaknya efisien, karena menghitung wilayah unik untuk setiap i, tetapi berhasil.
```

```
tappl(auterr,"mean",austates,incomes)
[44.5, 57.3333, 55.5, 53.6, 55, 60.5, 56, 52.25]
```

Perhatikan bahwa ini berfungsi untuk setiap vektor wilayah.

setiap i, tetapi berhasil.

```
tappl(["act", "nsw"], "mean", austates, incomes)
[44.5, 57.3333]
```

Sekarang, paket statistik EMT mendefinisikan tabel seperti di R. Fungsi `readtable()` dan `writetable()` dapat digunakan untuk input dan output.

Jadi kita bisa mencetak rata-rata pendapatan negara di wilayah dengan cara yang 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
```

Kita juga dapat mencoba meniru perilaku R sepenuhnya.

Faktor-faktor tersebut harus dengan jelas disimpan dalam kumpulan dengan jenis dan kategori (negara bagian dan teritori dalam contoh kami). 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/pre` `statef=makef(austates);`

```
Sekarang elemen ketiga dari koleksi akan berisi indeks
statef[3]
```

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

Sekarang kita bisa meniru `tapply()` dengan cara berikut. Ini akan mengembalikan tabel sebagai kumpulan

data tabel dan judul kolom.

```
function tappl (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; end; return x, u, f; endfunction < /pre >
```

*Kami tidak menambahkan banyak jenis pengecekan disini. Satu-satunya tindakan pencegahan menyangkut kata*

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 data ini disebut matriks. Akan mudah untuk menulis fungsi untuk dimensi yang lebih tinggi atau pustaka C untuk ini.

R memiliki lebih dari dua dimensi. Dalam R, array adalah vektor dengan bidang dimensi.

Dalam EMT, vektor adalah matriks dengan satu baris. Vektor 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
```

Ekstraksi baris dan kolom, atau sub-matriks, sangat mirip dengan R.

```
X[,2:3]
```

```
2 3 7 8 12 13 17 18
```

Namun, dalam R dimungkinkan untuk menetapkan daftar indeks spesifik dari vektor ke suatu nilai. Hal yang sama dimungkinkan di EMT hanya dengan loop.

```
function setmatrixvalue(M, i, j, v) ...
```

```
loop 1 to max(length(i),length(j),length(v)) M[i,j] = v; end; endfunction
```

*/pre Kami mendemonstrasikan ini untuk menunjukkan bahwa matriks dilewatkan dengan referensi di EMT. Jika Anda tidak ingin mengubah matriks asli M, 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 menjadi vektor kolom dan yang lainnya vektor baris

```
(1:5)(1:5)'
```

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

Dalam pengantar PDF untuk R ada sebuah 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, ini dapat dicapai dengan loop. Tapi loop tidak efektif di EMT atau R. Di EMT, kita bisa menulis loop di C dan itu akan menjadi solusi tercepat.

Tapi kita ingin meniru perilaku R. Untuk ini, 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/23030630036_Marcelline
```

Selain multiplisitas yang tepat, EMT dapat menghitung frekuensi dalam vektor

```

getfrequencies(q,-50:10:50)
[0, 23, 132, 316, 602, 801, 333, 141, 53, 0]

```

Cara paling mudah untuk memplot ini sebagai distribusi adalah sebagai berikut.

```

plot2d(q,distribution=11):
!images/23030630036_Marcelline

```

Tetapi juga memungkinkan untuk menghitung sebelumnya hitungan 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.

tau R. Di EMT, kita bisa menulis loop di C dan itu akan menjadi solusi tercepat.

Tapi kita ingin meniru perilaku R. Untuk ini, kita perlu meratakan perkalian ab dan membuat matriks ab-cd.

```

x,y=histo(q,v=-55:10:55); y=y/sum(y)/differences(x); ... plot2d(x,y, bar,style="/" ):
!images/23030630036_Marcelline

```

Daftar

EMT memiliki dua jenis daftar. Satu adalah daftar global yang dapat diubah, dan yang lainnya adalah tipe daftar yang tidak dapat diubah. Kami tidak peduli dengan daftar global di sini.

Tipe daftar yang tidak dapat diubah disebut koleksi dalam EMT. Ia berperilaku seperti struktur dalam C, tetapi elemennya hanya diberi nomor dan tidak diberi nama.

```

L="Fred","Flintstone",40,[1990,1992]
Fred Flintstone 40 [1990, 1992]

```

Saat ini elemen tidak memiliki nama, meskipun nama dapat ditetapkan untuk tujuan khusus. Mereka diakses dengan angka.

```

(L[4])[2]
1992

```

File Input and Output (Membaca dan Menulis Data)

Anda akan sering ingin mengimpor matriks data dari sumber lain ke EMT. Tutorial ini memberi tahu Anda tentang banyak cara untuk mencapai ini. Fungsi sederhana adalah `writematrix()` dan `readmatrix()`.

Mari kita tunjukkan cara membaca dan menulis vektor real ke file.

```

a=random(1,100); mean(a), dev(a),
0.48264 0.29467

```

Untuk menulis data ke file, kita menggunakan fungsi `writematrix()`.

Karena pengenalan ini kemungkinan besar berada di direktori, di mana pengguna tidak memiliki akses tulis, kami menulis data ke direktori home pengguna. Untuk notebook sendiri, ini tidak perlu, 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, kami menggunakan readmatrix().
a=readmatrix(filename)';
Dan hapus file tersebut.
fileremove(filename);
mean(a), dev(a),
0.48264 0.29467
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 oleh titik
koma dalam file csv (defaultnya adalah nilai yang dipisahkan koma). File
"test.csv" berikut akan muncul di folder current Anda.
    filename="test.csv"; ... writematrix(random(5,3),file=filename,separator=",");
Anda sekarang dapat membuka file ini dengan Excel Indonesia secara lang-
sung
    fileremove(filename);
Terkadang kita butuh memiliki string dengan token seperti berikut ini.
s1:="f m m f m m m f f f m m f"; ... s2:="f f f m m f f";
Untuk tokenize 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();
Filanya akan terlihat seperti berikut.
printf(file)
A,B,C 0.2155037742656752,0.760609127418965,0.9712977947482432 0.08065595471102564,0.3823168531511
0.5063360766687275,0.801747417146587,0.925045846322969
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 notebook, atau ke file.
writetable(L,wc=10,dc=5)
A B C 0.2155 0.76061 0.9713 0.08066 0.38232 0.46045 0.50634 0.80175 0.92505
Matriks nilai adalah elemen pertama dari L. Perhatikan bahwa mean() dalam
EMT menghitung nilai rata-rata dari baris matriks.
mean(L[1])
0.64914 0.30781 0.74438
File CSV

```

Pertama, mari kita menulis matriks ke dalam file. Untuk output, kami membuat file di direktori kerja saat ini.

```
file="test.csv"; ... M=random(3,3); writematrix(M,file);
```

Berikut adalah isi dari file ini.

```
printfile(file)
```

```
0.8741133605593981,0.3260096198463895,0.6195246948206793 0.9568690589321172,0.8723714729545522,0.2
0.5198293315179149,0.3056519720612176,0.3469196197056061
```

CSV ini dapat dibuka pada sistem bahasa 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.

Tetapi titik desimal juga merupakan format default untuk EMT. Anda dapat membaca matriks dari file dengan `readmatrix()`.

```
readmatrix(file)
```

```
0.87411 0.32601 0.61952 0.95687 0.87237 0.27797 0.51983 0.30565 0.34692
```

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 mengeksport matriks sebagai CSV (nilai yang dipisahkan koma). Di Excel 2007, gunakan "simpan sebagai" dan "format lain", lalu pilih "CSV". Pastikan, tabel saat ini hanya berisi data yang ingin Anda ekspor.

Berikut adalah 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 Jerman saya menggunakan titik koma sebagai pemisah dan koma desimal. Anda dapat mengubah ini di pengaturan sistem atau di Excel, tetapi tidak perlu membaca matriks ke dalam EMT.

Cara termudah untuk membaca ini ke dalam Euler adalah `readmatrix()`. Semua koma diganti dengan titik dengan parameter `gt;comma`. Untuk CSV bahasa Inggris, cukup abaikan parameter ini

```
M=readmatrix("excel-data.csv", comma)
```

```
0 1000 1000 1 1051.3 1072.5 2 1105.2 1150.3 3 1161.8 1233.7 4 1221.4 1323.1
5 1284 1419.1 6 1349.9 1522 7 1419.1 1632.3 8 1491.8 1750.7 9 1568.3 1877.6 10
1648.7 2013.8
```

Mari gunakan plot ini.

```
plot2d(M'[1],M'[2:3], points,color=[red,green]):
```

```
![images/23030630036_Marcelline
```

Ada cara yang lebih mendasar untuk membaca data dari file. Anda dapat membuka file dan membaca angka baris demi baris. Fungsi `getvectorline()`

akan membaca angka dari baris data. Secara default, ia mengharapkan titik desimal. Tapi itu juga bisa menggunakan koma desimal, jika Anda memanggil `setdecimaldot(",")` sebelum Anda menggunakan fungsi ini.

Fungsi berikut adalah contoh untuk ini. Ini akan berhenti di akhir file atau baris kosong

```
function myload (file) ...
    open(file); M=[]; repeat until eof(); v=getvectorline(3); if length(v)~0 then
M=M_v; else break; end; end; return M; close(file); endfunction < /pre > myload(file)
0.87411 0.32601 0.61952 0.95687 0.87237 0.27797 0.51983 0.30565 0.34692
```

Dimungkinkan juga untuk membaca semua angka dalam file itu dengan `getvector()`.

```
open(file); v=getvector(10000); close(); redim(v[1:9],3,3)
0.87411 0.32601 0.61952 0.95687 0.87237 0.27797 0.51983 0.30565 0.34692
```

Jadi sangat mudah untuk menyimpan vektor nilai, satu nilai di setiap baris dan membaca kembali vektor ini.

```
v=random(1000); mean(v)
0.50841
writematrix(v',file); mean(readmatrix(file)')
0.50841
```

Menggunakan Tabel

Tabel dapat digunakan untuk membaca atau menulis data numerik. Misalnya, kita menulis tabel dengan tajuk baris dan kolom ke dalam sebuah berkas.

```
file="test.tab"; M=random(3,3); ... open(file,"w"); ... writetable(M,separator=",",labc=["one","two"],"
... close(); ... printfile(file)
```

```
one,two,three 0.98, 0.41, 0.73 0.1, 0.81, 0.78 0.99, 0.59, 0.97
```

Ini dapat diimpor ke dalam Excel.

Untuk membaca file dalam EMT, gunakan `readtable()`.

```
M,headings=readtable(file, clabs); ... writetable(M,labc=headings)
```

```
one two three 0.98 0.41 0.73 0.1 0.81 0.78 0.99 0.59 0.97
```

Menganalisis Garis

Anda bahkan dapat mengevaluasi setiap garis secara manual. Misalkan, kita memiliki garis dengan format berikut.

```
line="2020-11-03,Tue,1'114.05"
```

```
2020-11-03,Tue,1'114.05
```

Pertama kita dapat menandai garis.

```
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],"","")()
```

```
7.3816e+05 2 1114
```

Menggunakan ekspresi reguler, dimungkinkan untuk mengekstrak hampir semua informasi dari baris data.

Asumsikan kita memiliki baris berikut dokumen HTML.

```
line="jtr jtd 1145.45j/td jtd 5.6j/td jtd -4.5j/td jtr "
```

```

lt;trgt;lt;tdgt;1145.45lt;/tdgt;lt;tdgt;5.6lt;/tdgt;lt;tdgt;-4.5lt;/tdgt;lt;trgt;
Untuk mengekstrak ini, kami menggunakan ekspresi reguler, yang mencari
* kurung tutup gt;,
* string apa pun yang tidak mengandung tanda kurung dengan * sub-
pertandingan "(...)",
* bracket pembuka dan penutup menggunakan solusi terpendek,
* lagi string apa pun yang tidak mengandung tanda kurung,
* dan kurung buka lt;.
Ekspresi reguler agak sulit dipelajari tetapi sangat kuat.
pos,s,v=strxfind(line,"([< ]+)< .+? ([< ]+)< ");
Hasilnya adalah posisi kecocokan, string yang cocok, dan vektor string untuk
sub-pertandingan.
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,v=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; readtd(line+"jtd non-numericali/td ")
1145.45 5.6 -4.5 non-numerical
Membaca dari Web
Situs web atau berkas dengan URL dapat dibuka di EMT dan dapat dibaca
baris demi baris.
Dalam contoh ini, kami membaca versi terkini dari situs EMT. Kami meng-
gunakan ekspresi reguler untuk memindai "Version ..." dalam judul.
function readversion () ...
urlopen("http://www.euler-math-toolbox.de/Programs/Changes.html"); re-
peat until urloef(); s=urlgetline(); k=strfind(s,"Version ",1); if k>0 then substring(s,k,strfind(s,"j",k)-
1); break; endif; end; urfclose(); endfunction j/pre; readversion
Version 2024-01-12
Variabel Input dan Output
Anda dapat menulis sebuah variabel dalam bentuk definisi Euler ke file atau
ke baris perintah.
writevar(pi,"mypi");
mypi = 3.141592653589793;
Untuk pengujian, kami membuat file Euler di direktori kerja EMT
file="test.e"; ... writevar(random(2,2),"M",file); ... printfile(file,3)
M = [ .. 0.5440197834939144, 0.1655588178301693; 0.7665929189698761,
0.401290437257703];
Kita sekarang dapat memuat file. Ini akan mendefinisikan matriks M
load(file); show M,
M = 0.54402 0.16556 0.76659 0.40129
Omong-omong, jika writevar() digunakan pada variabel, itu akan mencetak
definisi variabel dengan nama variabel ini.
writevar(M); writevar(inch)

```

```
M = [ .. 0.5440197834939144, 0.1655588178301693; 0.7665929189698761,
0.401290437257703]; inch= 0.0254;
```

Kita juga bisa membuka file baru atau menambahkan file yang sudah ada. Dalam contoh kami menambahkan file yang dihasilkan sebelumnya.

```
open(file,"a"); ... writevar(random(2,2),"M1"); ... writevar(random(3,1),"M2");
... close();
```

```
load(file); show M1; show M2;
```

```
M1 = 0.0087682 0.65039 0.40482 0.2308 M2 = 0.27018 0.42392 0.32925
```

Untuk menghapus file apa pun gunakan `fileremove()`.

```
fileremove(file);
```

Vektor baris dalam file tidak memerlukan koma, jika setiap angka berada di baris baru. Mari kita buat file seperti itu, menulis setiap baris satu per satu dengan `writeln()`.

```
open(file,"w"); writeln("M = ["); ... for i=1 to 5; writeln(" "+random());
end; ... writeln("];"); close(); ... printfile(file)
```

```
M = [ 0.441957682635 0.553673438291 0.130434088915 0.676476042951 0.437161301337
];
```

```
load(file); M
```

```
[0.44196, 0.55367, 0.13043, 0.67648, 0.43716]
```

Latihan

1. Buatlah boxplot dari data berikut ini dan sertakan penjelasan dari boxplot tersebut (kuartil, median, nilai max/min) dari

```
P = 90,100,85,87,98,99,34,59,29,96,87,85,83,80
```

```
P=[90,100,85,87,98,99,34,59,29,96,87,85,83,80]; boxplot(P):
```

```
![images/23030630036_Marcelline
```

2. Diberikan data dari penjualan kendaraan pada tahun 2045.

```
Januari = 100
```

```
Februari = 200
```

```
Maret = 250
```

```
April = 50
```

```
Mei = 300
```

```
Juni = 500
```

```
Juli = 680
```

```
Agustus = 300
```

```
September = 450
```

```
Oktober = 450
```

```
November = 600
```

```
Desember = 700
```

```
month=["Jan","Feb","Mar","Apr","Mei","Juni","Juli","Agt","Sept","Okt","Nov","Des"];
values=[100,200,250,50,300,500,680,300,450,450,600,700];
```

```
columnplot(values,lab=month,color=green);
```

```
title("Data Penjualan Kendaraan Tahun 2045");
```

```
![images/23030630036_Marcelline
```

3. Dengan soal yang sama, buatlah diagram lingkaran



```

CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.9,0,0)]
[5.87532e+07, 2, 15, 3, 6.54049e+07]
i=[1,2,3,4,5,6,7,8,9,10,11,12]; piechart(values[i],color=CP[i],lab=month[i]:
Index 6 out of bounds! Error in: ... 6,7,8,9,10,11,12]; piechart(values[i],color=CP[i],lab=month[i]
...
—
4. Dengan soal yang sama, buatlah diagram star!
aspect(2.5); starplot(values,lab=month, rays):
!images/23030630036_Marcelline
—
5. Uji Chi-Kuadrat
Penelitian dilakukan untuk mengetahui partisipasi warga dalam suatu kepala
desa yang dilihat dari jenis kelamin. Hasil penelitian tersebut diperoleh data
sebagai berikut
Jenis Kelamin — Ikut — Tidak
Pria — 10 — 5
Wanita — 9 — 6
Apakah ada pengaruh jenis kelamin terhadap keikutsertaan dalam pemilihan
kepala desa?
votingGenders = [10,5;9,6]
10 5 9 6
//bentuk menjadi sebuah tabel
writetable(votingGenders, wc=10, labr=["Pria","Wanita"], labc=["Ikut","Tidak"])
Ikut Tidak Pria 10 5 Wanita 9 6
tabletest(votingGenders) //periksa tabel untuk uji chi-kuadrat
0.704786173865
karena nilai yang terlalu besar, maka belum dapat disimpulkan apakah ter-
dapat hubungan antara keikutsertaan dengan gender.
Lalu, dibuat tabel nilai harapan untuk melihat nilai harapan pada tabel
tersebut
writetable(expectedtable(votingGenders), wc=10, labr=["Pria","Wanita"],
labc=["Ikut","Tidak"])
Ikut Tidak Pria 9.5 5.5 Wanita 9.5 5.5

```