

TUGAS APLIKASI KOMPUTER

KB 15-16: PENGGUNAAN MARKDOWN DAN LATEX



Oleh:
SHINTIA NURCHOLISA
23030630071

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

BAB I

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

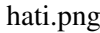
[F11] lagi untuk kembali ke tata letak favorit Anda. Tata letak disimpan untuk sesi berikutnya.

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 -> Edit Comment atau tekan [F5], dan juga baris perintah EMT yang ditandai dengan “>” dan berwarna merah. Anda dapat menyisipkan baris perintah baru dengan cara menekan tiga tombol bersamaan: [Shift]+[Ctrl]+[Enter].

1.1 Komentar (Teks Uraian)

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

- – Judul
- – Sub-Judul
- latex: $F(x) = \int \hat{a} x f(t), dt$ $\mathit{mathjax} : \frac{x^2-1}{x-1} = x + 1$
- maxima: $\text{'integrate}(x^3,x) = \text{integrate}(x^3,x) + C$
- <http://www.euler-math-toolbox.de>
- See: <http://www.google.de> | Google
- image: 
- —

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

BAB II

JUDUL

2.1 Sub-Judul

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

maxima: 'integrate(x^3,x) = integrate(x^3,x) + C

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

BAB III

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
```

3.1 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.

```
>(785-3)^2-1000
```

```
610524
```

>200-3+77

274

>55+6*3

73

>O=12749; 48^2-O

-10445

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

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

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

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

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

1.25

Argumen atau input untuk fungsi ditulis di dalam tanda kurung.
>sin(45°), cos(pi), log(sqrt(E))

0.707106781187

-1

0.5

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

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

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

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

1.61803398875

2

3.2 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.

>sin(35°)

0.573576436351


```
>log10(400)
```

```
2.60205999133
```

```
>cot(45)
```

```
0.617369623784
```

```
>mod(45,3)
```

```
0
```

```
>ceil(0.012234465657)
```

```
1
```

BAB IV

SATUAN

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

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

1609.344

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

```
kilometer$:=1000;
km:= kilometer;
cm$:=0.01;
mm$:=0.001;
minute$:=60;
min:= minute;
minutes:= minute;
hour:= 60 * minute;
h:= hour;
hours:= hour;
day:= 24 * hour;
days:= day;
d:= day;
year:= 365.2425 * day;
years:= year;
y:= year;
inch$:=0.0254;
in:= inch;
feet:= 12 * inch;
foot:= feet;
```

```

ft:= feet;
yard:= 3 * feet;
yards:= yard;
yd:= yard;
mile:= 1760 * yard;
miles:= mile;
kg$:=1;
sec$:=1;
ha$:=10000;
Ar$:=100;
Tagwerk$:=3408;
Acre$:=4046.8564224;
pt$:=0.376mm;

```

Untuk konversi ke dan antar unit, EMT menggunakan operator khusus, yakni ->.

>4km -> miles, 4inch -> " mm"

```

2.48548476895
101.6 mm

```

BAB V

FORMAT TAMPILAN NILAI

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

>pi

3.14159265359

>longest pi

3.141592653589793

>long pi

3.14159265359

>short pi

3.1416

>shortest pi

3.1

>fraction pi

312689/99532

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

1612.7

2.71828182846

3.141592653589793

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

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

>longestformat; pi, defformat; pi

3.141592653589793

3.14159265359

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

>1/7+1/4, fraction %

0.392857142857

11/28

BAB VI

PERINTAH MULTIBARIS

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

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

```
>a=4; b=15; c=2; // menyelesaikan  $a \cdot x^2 + b \cdot x + c = 0$  secara
manual ...
> D=sqrt(b2/(a*2*4)-c/a); ...
> -b/(2*a) + D, ...
> -b/(2*a) - D

-0.138444501319
-3.61155549868
```


BAB VII

MENAMPILKAN DAFTAR VARIABE

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

```
>listvar
```

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

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

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

Contoh berikut ini menampilkan semua unit yang diakhiri dengan “m” dengan mencari semua variabel yang berisi “m\$”.

```
>listvar m$
```

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

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


```
>remvalue a,b,c,D
```

```
>D
```

```
Variable D not found!
```

```
Error in:
```

```
D ...
```

```
^
```

BAB VIII

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  
0.193585, 0.463387, 0.095153, 0.595017]
```

```
>normal(10)
```

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


BAB IX

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, 1.0]
```

```
>mean(w^2)
```

```
0.35
```


BAB X

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).

$\text{re}(x)$: bagian riil pada bilangan kompleks x .

$\text{im}(x)$: bagian imajiner pada bilangan kompleks x .

$\text{complex}(x)$: mengubah bilangan riil x menjadi bilangan kompleks.

$\text{conj}(x)$: Konjugat untuk bilangan bilangan komplkes x .

$\text{arg}(x)$: argumen (sudut dalam radian) bilangan kompleks x .

$\text{real}(x)$: mengubah x menjadi bilangan riil.

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

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

```
>sqrt(complex(-1))
```

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

```
2+3i
```

```
2
```

```
3
```

```
2-3i
```

```
0.982793723247
```

```
56.309932474
```

```
56.309932474
```

```
>deg(arg(I)) // 90°
```

```
90
```

```
>sqrt(-1)
```

```
Floating point error!
```

```
Error in sqrt
Error in:
sqrt(-1) ...
      ^
```

```
>sqrt(complex(-1))
```

```
0+1i
```

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

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

```
>complex(-1), sqrt(%)
```

```
-1+0i
```

```
0+1i
```

BAB XI

MATEMATIKA SIMBOLIK

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

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

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

```
>&(a+b)^2
```

$$(b + a)^2$$

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

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

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

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

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

`>(a2-b2)/(a+b), &ratsimp(%) // ratsimp menyederhanakan bentuk pecahan`

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

$$a - b$$

`>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)`

$$\begin{matrix} 3 & 3 \\ 2 & 5 \end{matrix}$$

`>:: factor(20!)`

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

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

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

10

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

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

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

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

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

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

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

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

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

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

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

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

$$x^2 - x + 1$$

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

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

7

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

$$\frac{(b + a)^3 + 1}{b + a + 1}$$

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

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

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

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

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

BAB XII

TAMPILAN MATEMATIKA SIMBOLIK DENGAN LATEX

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

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

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

$$(b + a)^2$$

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

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

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

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

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

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

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

$$a - b$$

Selamat Belajar dan Berlatih!

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

Sebagai latihan untuk memperlancar penggunaan perintah-perintah EMT yang sudah dijelaskan di atas, silakan Anda lakukan hal-hal sebagai berikut.

- Carilah soal-soal matematika dari buku-buku Matematika.
- Tambahkan beberapa baris perintah EMT pada notebook ini.
- Selesaikan soal-soal matematika tersebut dengan menggunakan EMT.
- Pilih soal-soal yang sesuai dengan perintah-perintah yang sudah dijelaskan dan dicontohkan di atas.

• _____

1.tentukan turunan dari persamaan berikut

$$G = (5x^{7-2})^{(6x^2+2)}$$

$$(6x^2 + 2)^7 (5x^5 - 2)$$

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

$$12x^7 (5x^5 - 2) + 35x^6 (6x^2 + 2)$$

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

$$K = 4x^3$$

$$4x^3$$

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

$$81$$

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

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

$$- 3 \sin(1)$$

4. Tentukan turunan ketiga dari fungsi berikut

$$>N \&= x^{3-3*x}2+8$$

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

$$>M \&= \text{diff}(N,x) \text{ //turunan pertama dari } N$$

$$3x^2 - 6x$$

$$>T \&= \text{diff}(M,x) \text{ //turunan kedua dari } N$$

$$6x - 6$$

$$>\&\text{diff}(T,x) \text{ //turunan ketiga dari } N$$

$$6$$

unicodehyperref hyphensurl []book xcolor amsmath,amssymb
iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype
[protrusion]basicmath parskip bookmark xurl

EMT untuk Perhitungan Aljabar

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

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

Contoh pertama

Menyederhanakan bentuk aljabar:

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

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

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

Menjabarkan:

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

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

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

Baris Perintah

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

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

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

```
16.7551608191
```

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

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

```
50.2654824574
```

```
100.530964915
```

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

```
>x := 1;
```

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

```
0.540302305868
```

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

0.857553215846

Jika dua baris dihubungkan dengan “...” kedua baris akan selalu dijalankan secara bersamaan.

```
>x := 1.5; ...  
> x := (x+2/x)/2, x := (x+2/x)/2, x := (x+2/x)/2,
```

```
1.416666666667  
1.41421568627  
1.41421356237
```

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

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

```
>%+ x=4+5; ...  
> //Baris ini tidak akan terlihat setelah kursor adalah dari baris
```

Garis yang dimulai dengan %% akan benar-benar tidak terlihat.

81

Euler mendukung loop dalam baris perintah, asalkan sesuai dengan satu baris atau beberapa baris. Dalam program, pembatasan ini tentu saja tidak berlaku. Untuk informasi lebih lanjut, lihat pengantar berikut.

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

```
1.5  
1.416666666667  
1.41421568627  
1.41421356237  
1.41421356237
```

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

```
>x := 1.5; //komentar diletakkan di sini sebelum ...  
> repeat xnew:=(x+2/x)/2; until xnew~=x; ...  
> x := xnew; ...  
> end; ...  
> x,
```

Struktur bersyarat juga berfungsi.

```
>if Epi>piE; then "Thought so!"; endif;
```

Thought so!

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

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

```
>sqrt(sin(10°)/cos(20°))
```

0.429875017772

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

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

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

2.5

Sintaks Dasar

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

Untuk menyetel variabel, gunakan “=” atau “:=”. Demi kejelasan, pendahuluan ini menggunakan bentuk yang terakhir. Spasi tidak penting. Tapi jarak antar perintah diharapkan.

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

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

```
30.65625
```

EMT menggunakan sintaks pemrograman untuk ekspresi. Memasuki

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

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

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

```
8.77908249441
```

Untuk menghitung ekspresi yang rumit seperti

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

Anda harus memasukkannya dalam bentuk baris.

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

23.2671801626

Tempatkan tanda kurung dengan hati-hati di sekitar sub-ekspresi yang perlu dihitung terlebih dahulu. EMT membantu Anda dengan menyoroti ekspresi bahwa kurung tutup selesai. Anda juga harus memasukkan nama “pi” untuk huruf Yunani pi.

Hasil dari perhitungan ini adalah bilangan floating point. Ini secara default dicetak dengan akurasi sekitar 12 digit. Di baris perintah berikut, kita juga belajar bagaimana kita bisa merujuk ke hasil sebelumnya dalam baris yang sama.

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

0.47619047619

10/21

Perintah Euler bisa berupa ekspresi atau perintah primitif. Ekspresi dibuat dari operator dan fungsi. Jika perlu, itu harus berisi tanda kurung untuk memaksa urutan eksekusi yang benar. Jika ragu, menetapkan braket adalah ide yang bagus. Perhatikan bahwa EMT menampilkan tanda kurung buka dan tutup saat mengedit baris perintah.

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

14.4978445072

Operator numerik Euler termasuk

- unary atau operator plus
 - unary atau operator minus
 - *, /
 - produk matriks
 - a^b pangkat untuk positif a atau bilangan bulat b ($a^{**}b$ bekerja juga)
 - $n!$ operator faktorial
- dan masih banyak lagi.

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

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

Beberapa perintah memiliki alias, mis. ln untuk log.

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

```
2  
0.5
```

```
>sin(30°)
```

```
0.5
```

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

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

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

Bilangan Real

Tipe data utama di Euler adalah bilangan real. Real direpresentasikan dalam format IEEE dengan akurasi sekitar 16 digit desimal.

```
>longest 1/3
```

```
0.3333333333333333
```

Representasi ganda internal membutuhkan 8 byte.

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


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

```
>v:=[“affe”,“charlie”,“bravo”]
```

```
affe  
charlie  
bravo
```

Vektor string kosong dilambangkan dengan [tidak ada]. Vektor string dapat digabungkan.

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

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

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

String unicode dapat digabungkan seperti string lainnya.

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

```
= 45°
```

I

Dalam komentar, entitas yang sama seperti, dll. Dapat digunakan. Ini mungkin alternatif cepat untuk Latex. (Keterangan lebih lanjut pada komentar di bawah).

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

```
>v=strtochar(u”Ä is a German letter”)
```

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

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

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

Ü is a German letter

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

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

We have =.

Dimungkinkan juga untuk menggunakan entitas numerik.

```
>u"Ähnliches"
```

Ähnliches

Nilai Boolean

Nilai Boolean diwakili dengan 1 = true atau 0 = false di Euler. String dapat dibandingkan, seperti halnya angka.

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

0

1

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

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

1

Operator Boolean mematuhi aturan bahasa matriks.

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

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

Anda dapat menggunakan fungsi `nonzeros()` untuk mengekstrak elemen tertentu dari vektor. Dalam contoh, kami menggunakan `fileisprime` bersyarat (`n`)

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

```
[2, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23,
31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51,
59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79,
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,
53, 59, 61, 67, 71, 73, 79, 83, 89, 97]
```

Format Keluaran

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

```
>defformat; pi
```

```
3.14159265359
```

Secara internal, EMT menggunakan standar IEEE untuk bilangan ganda dengan sekitar 16 digit desimal. Untuk melihat secara lengkap jumlah digit, gunakan perintah “`longestformat`”, atau kita menggunakan operator “terpanjang” untuk menampilkan hasilnya format terpanjang.

```
>longest pi
```

```
3.141592653589793
```

Berikut adalah representasi heksadesimal internal dari bilangan ganda

```
>printhex(pi)
```

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

Format keluaran dapat diubah secara permanen dengan perintah format.

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

```
0.33333
```

```
3.14159
```

```
0.84147
```

defaultnya adalah format (12).

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

```
0.333333333333
```

Fungsi seperti “shortestformat”, “shortformat”, “longformat” bekerja untuk vektor dengan cara berikut.

```
>shortestformat; random(3,8)
```

0.66	0.2	0.89	0.28	0.53	0.31	0.44	0.3
0.28	0.88	0.27	0.7	0.22	0.45	0.31	0.91
0.19	0.46	0.095	0.6	0.43	0.73	0.47	0.32

Format default untuk skalar adalah format (12). Tapi ini bisa diubah.

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

```
3.1416
```

Fungsi “format terpanjang” mengatur format skalar juga.

```
>longestformat; pi
```

```
3.141592653589793
```

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

```
shortestformat shortformat longformat, longestformat  
format(panjang,digit) goodformat(panjang)  
fracformat(panjang)  
defformat
```

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

Tetapi format keluaran EMT dapat diatur dengan cara yang fleksibel.

```
>longestformat; pi,
```

```
3.141592653589793
```

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

```
3.14159
```

defaultnya adalah defformat().

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

Ada operator pendek yang hanya mencetak satu nilai. Operator “terpanjang” akan mencetak semua digit valid dari jumlah.

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

```
4.934802200544679
```

Ada juga operator singkat untuk mencetak hasil dalam format pecahan. Kami telah menggunakannya di atas.

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

```
25/12
```

Karena format internal menggunakan cara biner untuk menyimpan angka, nilai 0.1 tidak akan direpresentasikan dengan tepat. Itukesalahan 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 “longformat” default Anda tidak akan melihat ini.
Untuk kenyamanan, keluaran angka sangat kecil adalah 0.

$>0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1+0.1-1$

0

EKSPRESI

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

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

```
12.56637061435917
```

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

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

```
-0.919535764538
```

Perhatikan bahwa ekspresi akan selalu menggunakan variabel global, meskipun ada variabel dalam fungsi yang samanama. (Jika tidak, evaluasi ekspresi dalam fungsi dapat memberikan hasil yang sangat membingungkan bagi pengguna yang disebut fungsi.)

```
>at:=4; function f(expr,x,at) := expr(x); ...
```

```
> f("at*x^2",3,5) // computes 4*3^2 not 5*3^2
```

```
36
```

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

```
>at:=4; function f(expr,x,a) := expr(x,at=a); ...
```

```
> f("at*x^2",3,5)
```

```
45
```


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

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

45

Ekspresi dalam x sering digunakan seperti fungsi. Perhatikan bahwa mendefinisikan fungsi dengan nama yang sama seperti ekspresi simbolik global menghapus variabel ini menjadikannya kebingungan antara ekspresi dan fungsi simbolik.

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

12

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

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

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

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

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

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

41

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 simbolis atau

numerik.sion. Jika variabel utamanya adalah x , ekspresi tersebut dapat dievaluasi seperti fungsi.

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

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

-0.475

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

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

-0.425

Ekspresi tidak perlu simbolis. Ini diperlukan, jika ekspresi berisi fungsi, yang hanyadikenal di kernel numerik, bukan di Maxima.

MATEMATIKA SIMBOLIK

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

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

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

```
>$&33!
```

```
8683317618811886495518194401280000000
```

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

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

```
>$& 33!/(34!*10!) // nilai C(33,10)
```

$$\frac{1}{123379200}$$

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

```
>$binomial(33,10)//menghitung C(33,10) menggunakan fungsi binomial()
```

```
92561040
```

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

Anda akan belajar bahwa yang berikut ini juga berfungsi

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

>\$binomial(x,4) // C(x,4)

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

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

>\$&binomial(x,7) with x=20 // substitusi x=20 ke C(x,7)

77520

Dengan begitu, Anda bisa menggunakan solusi persamaan di persamaan lain.

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

Seperti yang akan Anda lihat pada contoh sebelumnya dan berikut, jika Anda menginstal LaTeX, Anda dapat mencetak simbolekspresi bolic dengan Latex. Jika tidak, perintah berikut akan mengeluarkan pesan kesalahan.

Untuk mencetak ekspresi simbolik dengan LaTeX, gunakan \$ infront dari & (atau Anda dapat menghapus &) sebelum perintah. jangan menjalankan perintah Maxima dengan \$, jika Anda belum menginstal LaTeX

>\$(4+x)/(x^2+5)

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

Ekspresi simbolik diurai oleh Euler. Jika Anda membutuhkan sintaks kompleks dalam satu ekspresi, Anda dapat menyertakan ekspresi dalam “...”. Menggunakan lebih dari sekedar ekspresi sederhana dimungkinkan, tetapi sangat tidak disarankan.

>&“v := 7; v^2”

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

```
>$expand((1+x)^2), $&factor(diff(% ,x)) // diff: turunan,
factor: faktor
```

$$x^2 + 2x + 1$$

$$2(x + 1)$$

Sekali lagi, % mengacu pada hasil sebelumnya. Untuk mempermudah, kami menyimpan solusi ke variabel simbolik. Variabel simbolik didefinisikan dengan “&=”.

```
>fx &= (x+5)/(x^5+5); $&fx
```

$$\frac{x + 5}{x^5 + 5}$$

Ekspresi simbolik dapat digunakan dalam ekspresi simbolik lainnya.

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

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

Masukan langsung dari perintah Maxima juga tersedia. Mulai baris perintah dengan “::”. Sintaks Maximadiadaptasi ke sintaks EMT (disebut “mode kompatibilitas”).

```
>&factor(25!)
```

```
15511210043330985984000000
```

```
>:: factor(15!)
```

$$\begin{array}{ccccccc} 11 & 6 & 3 & 2 & & & \\ 2 & 3 & 5 & 7 & 11 & 13 & \end{array}$$

```
>:: factor(50!)
```

47 22 12 8 4 3 2 2 2
2 3 5 7 11 13 17 19 23 29 31 37 41 43

Jika Anda ahli dalam Maxima, Anda mungkin ingin menggunakan sintaks asli Maxima. Anda dapat melakukan ini dengan “::”.

```
>:: av:g$ av^4;
```

4
g

```
>fx &= x^4*exp(x), $fx
```

4 x
x E

$x^4 e^x$

Variabel semacam itu dapat digunakan dalam ekspresi simbolik lainnya. Perhatikan, bahwa dalam perintah berikut ini tangan kanansisi & = dievaluasi sebelum penugasan ke Fx.

```
>&(fx with x=7), $%, &float(%)
```

7
2401 E

$2401 e^7$

2633016.213386728

```
>fx(10)
```

220264657.948

Untuk evaluasi ekspresi dengan nilai variabel tertentu, Anda dapat menggunakan operator “dengan”.

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

```
>&(fx with x=20)-(fx with x=15), &float(%)
```

$$\frac{160000}{E^{20}} - \frac{50625}{E^{15}}$$

$$7.746093726108509e+13$$

`>$factor(diff(fx,x,5))`

$$(x^4 + 20x^3 + 120x^2 + 240x + 120)e^x$$

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

`>tex(fx)`

$$x^4 \backslash, e^{\{x\}}$$

Ekspresi simbolik dapat dievaluasi seperti ekspresi numerik.

`>fx(0.10)`

$$0.000110517091808$$

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

`>$&fx with x=3/2`

$$\frac{81}{16} e^{\frac{3}{2}}$$

Penugasan juga bisa bersifat simbolis.

`>$&fx with x=5+t`

$$(t+5)^4 e^{t+5}$$

Pemecahan perintah memecahkan ekspresi simbolik untuk variabel di Maxima. Hasilnya adalah vektor solusi.

`>$&solve(x^3+x=7,x)`

$$\left[x = \left(\frac{\sqrt{1327}}{2 \cdot 3^{\frac{3}{2}}} + \frac{7}{2} \right)^{\frac{1}{3}} \left(-\frac{\sqrt{3}i}{2} - \frac{1}{2} \right) - \frac{\frac{\sqrt{3}i}{2} - \frac{1}{2}}{3 \left(\frac{\sqrt{1327}}{2 \cdot 3^{\frac{3}{2}}} + \frac{7}{2} \right)^{\frac{1}{3}}}, x = \left(\frac{\sqrt{1327}}{2 \cdot 3^{\frac{3}{2}}} + \frac{7}{2} \right)^{\frac{1}{3}} \right]$$

Bandingkan dengan perintah numerik “selesaikan” di Euler, yang membutuhkan nilai awal, dan secara opsional targetnilai.

>solve(“x^3+x”,3,y=3)

1.21341166276

Nilai numerik dari solusi simbolik dapat dihitung dengan evaluasi hasil simbolik. Eulerakan membaca tugas x = dll. Jika Anda tidak memerlukan hasil numerik untuk perhitungan lebih lanjut Andadapat juga membiarkan Maxima menemukan nilai numerik.

>sol &= solve(x^4+2*x=1,x); \$&sol, sol(), \$&float(sol)

$$\left[x = -\frac{\sqrt{\frac{4\sqrt{3}\left(\frac{2\sqrt{43}}{3^{\frac{3}{2}}}+2\right)^{\frac{1}{6}}}{3\left(\frac{2\sqrt{43}}{3^{\frac{3}{2}}}+2\right)^{\frac{2}{3}}-4}} - \left(\frac{2\sqrt{43}}{3^{\frac{3}{2}}}+2\right)^{\frac{1}{3}} + \frac{4}{3\left(\frac{2\sqrt{43}}{3^{\frac{3}{2}}}+2\right)^{\frac{1}{3}}}}{2} - \frac{\sqrt{3\left(\frac{2\sqrt{43}}{3^{\frac{3}{2}}}+2\right)^{\frac{2}{3}}}}{2\sqrt{3}\left(\frac{2\sqrt{43}}{3^{\frac{3}{2}}}+2\right)} \right]$$

Floating point error! Error in sqrt Error in expression: [x = -

sqrt(4sqrt(3)(2sqrt(43)/3^(3/2)+2)(1/6)/sqrt(3(2sqrt(43)/3^(3/2)+2)(2/3)-4)-
(2sqrt(43)/3^(3/2)+2)(1/3)+4/(3(2sqrt(43)/3^(3/2)+2)(1/3)))/2-sqrt(3(2sqrt(43)/3^(3/2)+2)(2/3)-
4)/(2sqrt(3)(2sqrt(43)/3^(3/2)+2)(1/6)),x = sqrt(4sqrt(3)(2sqrt(43)/3^(3/2)+2)(1/6)/sqrt(3(2sqrt(43)/3^(3/2)+2)(2/3)-4)-
(2sqrt(43)/3^(3/2)+2)(1/3)+4/(3(2sqrt(43)/3^(3/2)+2)(1/3)))/2-sqrt(3(2sqrt(43)/3^(3/2)+2)(2/3)-4)/(2sqrt(3)(2sqrt(43)/3^(3/2)+2)(1/6)),x = sqrt(3(2sqrt(43)/3^(3/2)+2)(2/3)-4)/(2sqrt(3)(2sqrt(43)/3^(3/2)+2)(1/6))-sqrt(-4sqrt(3)(2sqrt(43)/3^(3/2)+2)(1/6)/sqrt(3(2sqrt(43)/3^(3/2)+2)(2/3)-4)-
(2sqrt(43)/3^(3/2)+2)(1/3)+4/(3(2sqrt(43)/3^(3/2)+2)(1/3)))/2,x =

$\text{sqrt}(-4\text{sqrt}(3)(2\text{sqrt}(43)/3^{(3/2)+2})(1/6)/\text{sqrt}(3(2\text{sqrt}(43)/3^{(3/2)+2})(2/3)-4)-$
 $(2\text{sqrt}(43)/3^{(3/2)+2})(1/3)+4/(3(2\text{sqrt}(43)/3^{(3/2)+2})(1/3)))/2+\text{sqrt}(3(2\text{sqrt}(43)/3^{(3/2)+2})$
 $4)/(2\text{sqrt}(3)(2\text{sqrt}(43)/3^{(3/2)+2})(1/6))]$ Error in: sol &= solve(x^4+2*x=1,x);
 \$sol, sol(), \$float(sol) ... ^

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

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

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

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

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

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

$$[[x = 2, y = 1], [x = 1, y = 2]]$$

$$2$$

Ekspresi simbolis dapat memiliki bendera, yang menunjukkan perlakuan khusus dalam Maxima. Beberapa bendera dapat digunakan sebagaiperintah juga, orang lain tidak bisa. Bendera ditambahkan dengan “|” (a nicer form of “ev(...,flags)”)

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

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

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

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

>\$factor(%)

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

Fungsi

Dalam EMT, fungsi adalah program yang ditentukan dengan perintah “fungsi”. Ini bisa menjadi fungsi satu baris atau multiline fungsi.

Fungsi satu baris dapat berupa numerik atau simbolik. Fungsi satu baris numerik ditentukan oleh “:=”.

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

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

```
>f(7)
```

```
49.4974746831
```

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

```
>f(0:0.5:5)
```

```
[0, 0.559017, 1.41421, 2.70416, 4.47214, 6.73146, 9.12.7402, 16.4924, 20.744, 25.4951]
```

Fungsi dapat diplot. Sebagai ganti ekspresi, kita hanya perlu memberikan nama fungsi.

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

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

```
0.786151377757
```

Secara default, jika Anda perlu menimpa fungsi built-in, Anda harus menambahkan kata kunci “overwrite”. Menimpa fungsi built-in berbahaya dan dapat menyebabkan masalah pada fungsi lain tergantung pada fungsinya.

Anda masih bisa memanggil fungsi built-in sebagai “...”, jika itu berfungsi di inti Euler.

```
>function overwrite sin (x) := _sin(x°) // redine sinus dalam
derajat
```

```
>sin(45)
```

```
0.707106781187
```

Lebih baik kita menghapus definisi ulang sin ini.

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

```
0.707106781187
```

Parameter Default

Fungsi numerik dapat memiliki parameter default.

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

Menghilangkan parameter ini menggunakan nilai default.

```
>f(10)
```

```
100
```

Menyetelnya menimpa nilai default.

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

```
125
```

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

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

```
80
```

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

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

```
>a=7; f(6)
```

```
252
```

Tetapi parameter yang ditetapkan menggantikan nilai global.

Jika argumen tidak ada dalam daftar parameter yang telah ditentukan sebelumnya, itu harus dideklarasikan dengan “: =”!

```
>f(7,a:=6)
```

294

Fungsi simbolik didefinisikan dengan “& =”. Mereka didefinisikan di Euler dan Maxima, dan bekerja di kedua dunia. Ekspresi yang menentukan dijalankan melalui Maxima sebelum definisi.

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

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

Fungsi simbolik dapat digunakan dalam ekspresi simbolik.

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

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

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

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

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

178.635099908

Mereka dapat digunakan untuk mendefinisikan fungsi atau ekspresi simbolik lainnya.

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

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

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

0.703467422498

Cara berikut juga berfungsi, karena Euler menggunakan ekspresi simbolik dalam fungsi g, jika tidak menemukan variabel simbolis g, dan jika terdapat fungsi simbolik g.

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

0.703467422498

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

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

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

$$(x + 2)^n$$

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

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

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

```
>P(7,4)
```

28561

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

$$x^3 - x$$

With = the function is symbolic, and can be used in other symbolic expressions.

>integrate(f(x),x)

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

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

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

yang tidak dapat dievaluasi secara simbolis.

Jika kita mendefinisikan ulang fungsi dengan kata kunci “map”, ini dapat digunakan untuk vektor x. Secara internal, fungsi ini dipanggil untuk semua nilai x satu kali, dan hasilnya disimpan dalam vektor.

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

>f(0:0.5:2)

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

Fungsi dapat memiliki nilai default untuk parameter.

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

Sekarang fungsi tersebut dapat dipanggil dengan atau tanpa parameter “base”.

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

2

6.7

Selain itu, dimungkinkan untuk menggunakan parameter yang ditetapkan.

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

2

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

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

$$b^2 - a b + b + a^2$$

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

Fungsi simbolik seperti itu dapat digunakan untuk variabel simbolik.

Tetapi fungsinya juga dapat digunakan untuk vektor numerik.

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

17

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

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

$$\text{diff}(\text{expr}, y, 2) + \text{diff}(\text{expr}, x, 2)$$


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

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

0

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

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

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

Untuk meringkas

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

MEMECAHKAN EKSPRESI

Ekspresi dapat diselesaikan secara numerik dan simbolik.

Untuk menyelesaikan ekspresi sederhana dari satu variabel, kita dapat menggunakan fungsi Solving (). Diperlukan nilai awal untuk memulai pencarian. Secara internal, Solving () menggunakan metode garis potong.

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

```
1.41421356237
```

Ini bekerja untuk ekspresi simbolik juga. Ambil fungsi berikut.

```
> %solve(x^2=4,x)
```

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

```
> %solve(x^4-4,x)
```

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

```
> %solve(a*x^4+b*x+c=10,x)
```

$$x = -\sqrt[4]{a \sqrt[3]{\frac{2\sqrt{3}b \left(\frac{\sqrt{-256ac^3+7680ac^2-76800ac+27b^4+256000a}}{23^{\frac{3}{2}}a^2} + \frac{b^2}{2a^2} \right)^{\frac{1}{6}}}{\frac{3a \left(\frac{\sqrt{-256ac^3+7680ac^2-76800ac+27b^4+256000a}}{23^{\frac{3}{2}}a^2} + \frac{b^2}{2a^2} \right)^{\frac{2}{3}}}{+4c-40}}}} - \left(\sqrt{-2} \right)$$

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

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

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

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

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

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

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

0.966715594851

2

Memecahkan ekspresi simbolis dalam bentuk simbolik mengembalikan daftar solusi. Kami menggunakan penyelesaian pemecah simbolik () yang disediakan oleh Maxima.

>sol &= solve(x²-x-1,x); \$&sol

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

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

>longest sol()

-0.6180339887498949

1.618033988749895

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

>\$&x² with sol[1], \$&expand(x²-x-1 with sol[2])

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

Sistem pemecahan persamaan secara simbolis dapat dilakukan dengan vektor persamaan dan penyelesaian pemecah simbolik (). Jawabannya adalah daftar persamaan.

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

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

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

$$a^x - x^a = 0, 1$$

dengan $a = 3$.

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

Salah 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

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

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

2.54116291558

MENYELESAIKAN PERTIDAKSAMAAN

Untuk menyelesaikan pertidaksamaan, EMT tidak akan dapat melakukan, melainkan dengan bantuan Maxima, yaitu 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
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 - 1 # 0],[x]) // x^2-1 <> 0
```

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

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

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

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

emptyset

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

universalset

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

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

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

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

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

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

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

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

>\$&fourier_elim((x + y < 5) and (x - y > 8),[x,y])

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

>\$&fourier_elim(((x + y < 5) and x < 1) or (x - y > 8),[x,y])

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

>&fourier_elim([max(x,y) > 6, x # 8, abs(y-1) > 12],[x,y])

[6 < x, x < 8, y < -11] or [8 < x, y
or [x < 8, 13 < y] or [x = y, 13 < y] or [8 < x
or [y < x, 13 < y]

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

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

Bahasa Matriks

Dokumentasi inti EMT berisi diskusi terperinci tentang bahasa matriks Euler.

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

>A=[5,4;7,6]

5	4
7	6

Produk matriks dilambangkan dengan titik.

>b=[5;8]

5
8

>b' // transpose b

[5, 8]

>inv(A) //inverse A

3	-2
-3.5	2.5

>A.b //perkalian matriks

57
83

>A.inv(A)

1	0
0	1

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

>A.A

53	44
77	64

>A^2 //perpangkatan elemen2 A

25	16
49	36

>A.A.A

573	476
833	692

>power(A,3) //perpangkatan matriks

573	476
833	692

>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)

1	0.8
0.875	0.75

>A\b // hasilkali invers A dan b, $A^{-1}b$

-1
2.5

>inv(A).b

-1
2.5

>A\A // $A^{-1}A$

1	0
0	1

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

1	0
0	1

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

25	16
49	36

Ini bukan hasil perkalian matriks, tetapi perkalian elemen dengan elemen. Pekerjaan yang sama untuk vektor.

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

25
64

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

```
>2*A
```

10	8
14	12

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

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

5	8
7	12

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

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

10	12
14	18

Dapat dibayangkan perkalian ini seolah-olah vektor baris v telah diduplikasi untuk membentuk matriks dengan ukuran yang sama dengan A.

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

1	2
1	2

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

5	8
7	12

Ini juga berlaku untuk dua vektor di mana satu adalah vektor baris dan yang lainnya adalah vektor kolom. Kami menghitung $i * j$ untuk i, j dari 1 sampai 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
2	4	6	8
3	6	9	12
4	8	12	16
5	10	15	20

Sekali lagi, ingatlah bahwa ini bukan hasil perkalian matriks!

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

55

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

55

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

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

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

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

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

5

Euler memiliki operator perbandingan, seperti “==”, yang memeriksa kesetaraan.

Kami mendapatkan vektor 0 dan 1, di mana 1 berarti benar.

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

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

Dari vektor seperti itu, “nonzeros” memilih elemen bukan nol.

Dalam hal ini, kami 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 dapat menggunakan vektor indeks ini untuk mendapatkan nilai t yang sesuai.

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

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

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

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

```
[4, 48, 95, 139, 147, 191, 238, 282, 290, 334, 343, 433, 477, 524, 568, 576, 620, 667, 711, 719, 763, 862, 906, 953, 997]
```

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

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

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

```
112
```

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

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

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

Ini mengembalikan indeks elemen, yang bukan nol.

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

```
0,4
```

1	4
2	1
2	2
3	2

Indeks ini dapat digunakan untuk mengatur elemen ke 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 menyetel elemen pada indeks ke entri beberapa matriks lainnya.

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

```
0.765761      0.401188      0.406347      -0.1269
-0.122404     -0.691673      0.495975      0.9528
0.548138     -0.483902      0.444255      0.5392
```

Dan dimungkinkan untuk mendapatkan elemen dalam vektor.

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

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

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

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

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

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

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

```
[0.765761, 0.952814, 0.548138]
```

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

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

```
[0.765761, 0.952814, 0.548138]
```

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

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

1	1
2	4
3	1

$[-0.765761, \quad -0.952814, \quad -0.548138]$

FUNGSI MATRIKS LAINNYA (BUILDING MATRIX)

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

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

1	2	3
1	2	3

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

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

0.032444	0.0534171	0.595713	0.5644
0.83916	0.175552	0.396988	0.835
0.0257573	0.658585	0.629832	0.7708

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

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

```
>A|1
```

0.032444	0.0534171	0.595713	0.5644
0.83916	0.175552	0.396988	0.835
0.0257573	0.658585	0.629832	0.7708

Dimungkinkan untuk membuat matriks vektor baris dan kolom.

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

1	2	3
1	2	3


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

1	1
2	2
3	3

Tujuan utamanya adalah untuk menafsirkan vektor ekspresi untuk vektor kolom.

```
>“[x,x^2]”(v')
```

1	1
2	4
3	9

Untuk mendapatkan ukuran A, kita bisa menggunakan fungsi-fungsi berikut.

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

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

Untuk vektor, ada length().

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

```
9
```

Ada banyak fungsi lain yang menghasilkan matriks.

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

1	1
1	1

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

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



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

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

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

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

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

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

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

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

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

```
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47]
[0, 5, 0, 6, 0, 0, 0, 7, 0, 8, 0]
[2, 3, 5, 7, 23, 29, 31, 37, 41, 43, 47]
```

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

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

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

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

Kami mulai dengan matriks identitas.

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

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

Kemudian kami mengatur diagonal bawah (-1) menjadi 1: 4.

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

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

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

Berikut adalah fungsi yang mengembalikan matriks tri-diagonal.

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

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

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

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

1	2	3
4	5	6
7	8	9

Sekarang kita bisa mengekstrak diagonal.

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

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

Misalnya. Kita dapat membagi matriks dengan diagonalnya. Bahasa matriks memperhatikan bahwa vektor kolom d diterapkan ke matriks baris demi baris.

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

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

VEKTORISASI

Hampir semua fungsi di Euler bekerja untuk matriks dan input vektor juga, jika memungkinkan.

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

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

```
[1, 1.41421, 1.73205]
```

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

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

Dengan ini dan operator titik dua `a:delta:b`, vektor nilai fungsi dapat dibuat 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

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

```
[0, 0.171, 0.288, 0.357, 0.384, 0.375, 0.336, 0.271, 0.099, 0, -0.099, -0.192, -0.273, -0.336, -0.375, -0.357, -0.288, -0.171, 0]
```

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

Misalnya, vektor kolom dikali vektor baris mengembang menjadi matriks, jika operator diterapkan. Berikut ini, `v'` adalah vektor yang dialihkan (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, ini sangat berbeda dari hasil perkalian matriks. Produk matriks dilambangkan dengan titik “.” di EMT.

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

55

Secara default, vektor baris dicetak dalam format kompak.

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

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

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

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

5

25

Untuk mentranpose matriks, kami menggunakan 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 masih merupakan vektor baris. Jadi $v' \cdot v$ berbeda dari $v \cdot v'$.

`>v'.v`

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

$v \cdot v'$ menghitung norma v kuadrat untuk vektor baris v . Hasilnya adalah vektor 1×1 , yang bekerja seperti bilangan real.

`>v.v'`

30

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

`>norm(v)^2`

30

Operator dan fungsi mematuhi bahasa matriks Euler.

Berikut adalah ringkasan aturannya.

- Fungsi yang diterapkan ke vektor atau matriks diterapkan ke setiap elemen.
- Seorang operator yang beroperasi pada dua matriks dengan ukuran yang sama diterapkan berpasangan ke elemen matriks.
- Jika kedua matriks memiliki dimensi yang berbeda, keduanya diperluas dengan cara yang masuk akal, sehingga memiliki ukuran yang sama.

Mis., Nilai skalar dikalikan vektor mengalikan nilai dengan setiap elemen vektor. Atau matriks dikalikan dengan vektor (dengan `*`, bukan.) Memperluas vektor ke ukuran matriks dengan menduplikasinya.

Berikut ini adalah kasus sederhana dengan operator `^`.

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

[1, 4, 9]

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

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

1	2	3
2	4	6
3	6	9

Perhatikan bahwa produk skalar menggunakan produk matriks, bukan *!

>v.v'

14

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

sum, prod menghitung jumlah dan produk dari baris
cumsum, cumprod melakukan hal yang sama secara kumulatif
menghitung nilai ekstrem dari setiap baris
extrema mengembalikan vektor dengan informasi ekstrem
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) eigenvalues

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

[1, 4, 9]

14

[1, 5, 14]

Operator: menghasilkan vektor baris spasi yang 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 ada operator “|” dan “-”.

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

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

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

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

6

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

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

6

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

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

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

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

Bentuk singkat dari: adalah menghilangkan indeks sepenuhnya.

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

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

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

```
>A{4}
```

4

Matriks juga bisa 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 kami 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	
315	5.49779	0.707107	-0.7071
360	6.28319	1	

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

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

Fungsi yang tidak bekerja untuk input vektor harus di-vectorisasi. Hal ini dapat dicapai dengan kata kunci “peta” dalam definisi fungsi. Kemudian fungsi tersebut akan dievaluasi untuk setiap elemen dari parameter vektor.

Integrasi numerik `integ()` hanya berfungsi untuk batas interval skalar. Jadi kita perlu melakukan vektorisasi.

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

Kata kunci “map” membuat fungsi menjadi vektor. Fungsi tersebut sekarang akan bekerja

```
untuk vektor angka.
```

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

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


SUB-MATRIKS DAN ELEMEN-MATRIKS

Untuk mengakses elemen matriks, gunakan notasi braketa.

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

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

Kita dapat mengakses baris matriks yang lengkap.

```
>A[2]
```

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

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

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

```
2
```

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

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

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

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

Di sini kita ingin baris pertama dan kedua dari A.

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

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

7	8	9
4	5	6
1	2	3

Trik indeks juga bekerja dengan kolom.

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

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

2	3
5	6
8	9

Untuk singkatan “:” menunjukkan semua indeks baris atau kolom.

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

3
6
9

Cara lainnya, biarkan indeks pertama kosong.

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

2	3
5	6
8	9

Kita juga bisa mendapatkan baris terakhir A.

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

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

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

>A[1,1]=4

4	2	3
4	5	6
7	8	9

Kami juga dapat menetapkan nilai ke baris A.

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

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

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

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

5	6	-1
7	8	6
7	8	9

Selain itu, beberapa pintasan diperbolehkan.

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

0	0	-1
0	0	6
7	8	9

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

>A[4]


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

SORTING AND SHUFFLING

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 asli. Ini dapat digunakan untuk menyusun ulang vektor lain dengan cara yang sama.

Mari kita mengacak vektor.

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

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

Indeks tersebut berisi urutan `v`.

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

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

Ini bekerja untuk vektor string juga.

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

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

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

```
a  
a  
aa
```

d
e
e

Seperti yang Anda lihat, posisi entri ganda agak acak.

```
>ind
```

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

Fungsi unik mengembalikan daftar elemen unik vektor yang diurutkan.

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

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

Ini bekerja untuk vektor string juga.

```
>unique(s)
```

a
aa
d
e

ALJABAR LINIER

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

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

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

```
    -4  
    4.5
```

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

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

```
8.790745908981989e-13
```

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

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

1	2	3
4	5	6
7	8	9

Determinan dari matriks ini adalah 0.

```
>det(A)
```

```
0
```


MATRIKS SIMBOLIK

Maxima memiliki matriks simbolis. Tentu saja, Maxima dapat digunakan untuk soal-soal aljabar linier 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
```

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

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

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

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

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

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

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

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

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

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

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

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

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

Untuk mengekstrak vektor eigen tertentu, perlu pengindeksan yang cermat.

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

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

[1, - 1]

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

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

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

Dalam ekspresi simbolik, gunakan dengan.

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

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

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

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

$$[1, a]$$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Misalnya, nilai eigen dari A dapat dihitung secara numerik.

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

$$[16.1168, \quad -1.11684, \quad 0]$$

Atau secara simbolis. Lihat tutorial tentang Maxima untuk detailnya.

```
>$eigenvalues((A?))
```

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

Nilai Numerik dalam Ekspresi simbolik

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

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

$$\begin{array}{cc} 1 & 3.14159 \\ 4 & 5 \end{array}$$

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

```
>$A
```

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

Untuk menghindarinya, ada fungsi “mxmset (variabel)”.

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

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

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

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

$$1.4142135623730950488016887242097_B \times 10^0$$

$$1.414213562373095$$

Ketepatan angka floating point besar dapat diubah.

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

```
3.1415926535897932384626433832795028841971693  
4592307816406286208998628034825342117068b0
```

Variabel numerik dapat digunakan dalam ekspresi simbolik apa pun yang menggunakan “(**var?**)”.

Perhatikan bahwa ini hanya diperlukan, jika variabel telah ditentukan dengan “:=” atau “=” sebagai variabel numerik.

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

```
−5.424777960769379
```

Demo - Suku Bunga

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

Asumsikan Anda memiliki modal awal 5000 (katakanlah dalam dolar).

```
>K=5000
```

```
5000
```

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

```
>K*1.03
```

```
5150
```

Euler akan memahami sintaks berikut juga.

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

```
5150
```

Tapi lebih mudah menggunakan faktornya

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

```
1.03
5150
```

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

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

```
6719.58189672
```

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

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

```
6719.58
```

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

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

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

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

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

```
Real 1 x 11 matrix
```

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

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

```
>short 0:10
```

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

Kemudian semua operator dan fungsi di Euler dapat diterapkan ke elemen vektor untuk elemen. Begitu

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

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

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

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

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

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

Mari kita bandingkan kedua hasil tersebut, dengan dan tanpa pembulatan.

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

```
1271.61
1271.6071
```

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

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

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

```
Real 1 x 11 matrix
```

```
5000.00      5150.00      5304.50      5463.64      .
```

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

```
>VKr'
```

```
5000.00
5150.00
5304.50
5463.64
```

5627.55
5796.38
5970.27
6149.38
6333.86
6523.88
6719.60

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

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

5150.00
5000.00 5150.00 5304.50

Anehnya, kita juga bisa menggunakan indeks vektor. Ingat bahwa 1: 3 menghasilkan vektor [1,2,3].

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

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

6719.60
6719.58

Perbedaannya sangat kecil.

MEMECAHKAN PERSAMAAN

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

```
>function onepay (K) := K*q+R
```

Kami tidak harus menentukan q atau R untuk definisi fungsi. Hanya jika kita menjalankan perintah, kita harus menentukan nilai-nilai ini. Kami memilih $R = 200$.

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

```
Real 1 x 11 matrix
```

```
5000.00      5350.00      5710.50      6081.82      .
```

Bagaimana jika kita menghapus jumlah yang sama setiap tahun?

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

```
Real 1 x 11 matrix
```

```
5000.00      4950.00      4898.50      4845.45      .
```

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

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

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

```
Real 1 x 51 matrix
```

```
5000.00      4950.00      4898.50      4845.45      .
```

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

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

48.00

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

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

Fungsi `iterate()` memiliki satu trik lagi. Ini bisa mengambil kondisi akhir sebagai argumen. Maka itu akan mengembalikan nilai dan jumlah iterasi.

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

-19.83

47.00

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

Ini adalah pertanyaan yang hanya bisa dijawab secara numerik. Di bawah ini, kami akan mendapatkan rumus yang diperlukan. Maka Anda akan melihat bahwa tidak ada rumus mudah untuk suku bunga. Tetapi untuk saat ini, kami bertujuan untuk solusi numerik.

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

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

Iterasinya sama seperti di atas

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

Apalagi kami hanya tertarik pada nilai terakhir. Jadi kami mengambil indeks `[-1]`.

Mari kita coba tes.

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

-19.83

Sekarang kita bisa menyelesaikan masalah kita.

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

3.15

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

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

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

-336.08

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

Solusi Simbolis untuk Masalah Suku Bunga

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

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

$$R + q K$$

Sekarang kita dapat mengulang ini.

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

Kami melihat sebuah pola. Setelah n periode yang kita miliki

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

Rumusnya adalah rumus jumlah geometris, yang dikenal dengan Maxima.

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


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

Ini agak rumit. Jumlahnya dievaluasi dengan bendera “simpsum” untuk mengurangnya menjadi hasil bagi.

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)
```

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

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

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

```
-19.82504734650985
```

```
-19.82504734652684
```

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

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

```
20.51
```

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

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

Asumsikan kita mendapat pinjaman sebesar K, dan membayar n pembayaran R (dimulai setelah tahun pertama) meninggalkan sisa utang Kn (pada saat pembayaran terakhir). Rumusnya jelas

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

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

Biasanya rumus ini diberikan dalam bentuk

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

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

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

Kita bisa mencari nilai R secara simbolis.

>\$&solve(equ,R)

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

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

Tentu saja, kami memiliki batasan berikut.

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

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

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

Persamaan ini juga bisa diselesaikan untuk n . Ini terlihat lebih bagus, jika kita menerapkan beberapa penyederhanaan padanya.

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

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

Aljabar Linear Elementer

Matriks

Membuat matriks dari persamaan linear & Menuliskannya di kode EMT

$$4v - 3w - 5x - 3y + z = 7$$

$$2v - w + 3x + 6y + 2z = -4$$

$$v + w + x + y + z = 1$$

$$-3v + 9w - 8x + 3y - z = 10$$

$$v + 5w - 3x + 6y - 3z = 9$$

>spla &= 4*v-3*w-5*x-3*y+1*z= 7; splb &= 2*v-1*w+3*x+6*y+2*z=-4; splc &= v+w+x+y+z= 1; ...

> spld &= -3*v+9*w-8*x+3*y-z= 10; sple &= v+5*w-3*x+6*y-3*z=9; \$&spla, \$&splb, \$&splc, \$&spld, \$&sple

$$\left[z - 3y - 5x - 3w + \frac{4}{j+1}, z - 3y - 5x - 3w + \frac{4}{j+2}, z - 3y - 5x - 3w + \frac{4}{j+3} \right]$$

$$\left[2z + 6y + 3x - w + \frac{2}{j+1}, 2z + 6y + 3x - w + \frac{2}{j+2}, 2z + 6y + 3x - w + \frac{2}{j+3} \right]$$

$$\left[z + y + x + w + \frac{1}{j+1}, z + y + x + w + \frac{1}{j+2}, z + y + x + w + \frac{1}{j+3} \right] = 1$$

$$\left[-z + 3y - 8x + 9w - \frac{3}{j+1}, -z + 3y - 8x + 9w - \frac{3}{j+2}, -z + 3y - 8x + 9w - \frac{3}{j+3} \right]$$

$$\left[-3z + 6y - 3x + 5w + \frac{1}{j+1}, -3z + 6y - 3x + 5w + \frac{1}{j+2}, -3z + 6y - 3x + 5w + \frac{1}{j+3} \right]$$

Mencari nilai-nilai variabel dari persamaan tersebut

>sspl &= solve([spla,splb,splc,spld,sple],[v,w,x,y,z]); \$&sspl

[]

Membuat Augmented Matriks dari persamaan diatas

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

4.00	-3.00	-5.00	-3.00	1.00
2.00	-1.00	3.00	6.00	2.00
1.00	1.00	1.00	1.00	1.00
-3.00	9.00	-8.00	3.00	-1.00
1.00	5.00	-3.00	6.00	-3.00

```
>B := [7;-4;1;10;9]
```

```
7.00  
-4.00  
1.00  
10.00  
9.00
```

Mencari nilai-nilai lain yang bisa di dapatkan dari persamaan tersebut

```
>det(A) // mencari nilai determinan
```

```
3717.00
```

```
>&echelon (A) // mencari nilai eselon baris
```

```
[      80143857 ]  
[ 1  ----- ]  
[      25510582 ]  
[                ]  
[ 0           1 ]
```

```
>&rank (A) // mencari rank dari suatu matriks
```

```
2
```

```
>kernel (A)
```

```
0.00  
0.00  
0.00  
0.00  
0.00
```

```
>A.kernel (A)
```

```
0.00
0.00
0.00
0.00
0.00
```

>invert (A)// mencari invers dari matriks tersebut

```
[ - 0.6608188066430674    0.4152047016607668 ]
[
[ 0.5286550453144538    - 0.1321637613286135 ]
```

>A*B // mengalikan kedua matriks yang ada

```
28.00    -21.00    -35.00    -21.00    7.00
-8.00     4.00    -12.00    -24.00    -8.00
 1.00     1.00     1.00     1.00     1.00
-30.00    90.00    -80.00    30.00    -10.00
 9.00     45.00    -27.00    54.00    -27.00
```

>A.A // cross product terhadap matriks itu sendiri

```
15.00    -36.00    -13.00    -38.00    -7.00
-7.00     62.00    -64.00    21.00    -9.00
 5.00     11.00    -12.00    13.00     0.00
-12.00    14.00     13.00    58.00     7.00
-10.00    28.00    -32.00    24.00    11.00
```

>transpose (A)// mengtranspose matriks tersebut

```
4.00     2.00     1.00    -3.00     1.00
-3.00    -1.00     1.00     9.00     5.00
-5.00     3.00     1.00    -8.00    -3.00
-3.00     6.00     1.00     3.00     6.00
 1.00     2.00     1.00    -1.00    -3.00
```

Membuat Matriks Identitas

>C=id(4)

1.00	0.00	0.00	0.00
0.00	1.00	0.00	0.00
0.00	0.00	1.00	0.00
0.00	0.00	0.00	1.00

Vektor

Diberikan sebuah vektor dibawah ini, lalu dicari berbagai nilainya

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

```
1.00
1.00
1.00
```

```
>w:=[1;2;-1]
```

```
1.00
2.00
-1.00
```

```
>v:=[1;1;1]; w:=[1;2;-1]; v1:=w-(w'.v)/(v'.v)*v; fracprint(v1);
// mencari ortogonalitas vektor tersebut
```

```
1/3
4/3
-5/3
```

```
>v2:=crossproduct(v,w), // mencari nilai crossproduct
```

```
-3.00
2.00
1.00
```

```
>M:=[v/norm(v),v1/norm(v1),v2/norm(v2)] // mencari normalitas
dari vektor tersebut, seharusnya membentuk ortogonalitas
```

0.58	0.15	-0.80
0.58	0.62	0.53
0.58	-0.77	0.27

>sort([2,-1,1])// mengurutkan nilai vektor

-1.00	1.00	2.00
-------	------	------

MENCOBA LATIHAN SOAL

Faktorkan

$$z^2 - 81$$

`>&factor (z^2 - 81)// mengfaktorkan persamaan kuadrat`

$$(z - 9) (z + 9)$$

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

`>&factor (25*ab^4-25*az^4)`

$$- 25 (az - ab) (az + ab) (az^2 + ab^2)$$

$$t^2 - \frac{27}{100} + \frac{3}{5}t$$

`>&factor (t^2-27/100+3/5*t)`

$$\frac{(10 t - 3) (10 t + 9)}{100}$$

`>&load(fourier_elim)`

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

Selesaikan Pertidaksamaan

$$\left| \frac{2x + 1}{3} \right| > 5$$


```
>$&fourier.elim([abs(2*x+1)/3> 5],[x])
```

$$[7 < x] \vee [x < -8]$$

Selesaikan Persamaan tersebut menjadi suatu matriks yang diketahui nilai-nilainya.

$$4u + 4v - 5x - 3y + z = 7$$

$$3u + 2v - w + 3x + 6y = -4$$

$$6u + w + x + y + z = 1$$

$$-3v + 9w - 8x + 3y - z = 10$$

$$2u + v + 5w - 3x + 6y - 3z = 9$$

$$u + 4v + 3w - 2x + y + 5z = 4$$

```
>A := [4,4,0,-5,-3,1;3,2-1,3,0,6;6,0,1,1,1,1;0,-3,9,-8,3,1;2,1,5,-3,6,-3;1,4,3,-2,1]// Membuat Augmented Matriks dari persamaan diatas
```

```
Real 6 x 6 matrix
```

4.00	4.00	0.00	-5.00	...
3.00	1.00	3.00	0.00	...
6.00	0.00	1.00	1.00	...
0.00	-3.00	9.00	-8.00	...
2.00	1.00	5.00	-3.00	...
1.00	4.00	3.00	-2.00	...

```
>B := [7;-4;1;10;9;4]// Augmented Matriks dari hasil persamaan tersebut
```

7.00
-4.00
1.00
10.00
9.00
4.00

```

>spla &= 4*u+4*v+0-5*x-3*y+1*z= 7; splb &= 3*u+2*v-
1*w+3*x+6*y= -4; splc &= 6*u+w+x+y+z= 1; ...
> spld &= -3*v+9*w-8*x+3*y-z= 10; sple &= 2*u+v+5*w-3*x+6*y-
3*z=9; splf &= u+4*v+3*w-2*x+y+5*z=4; $&spla, $&splb, $&splc,
$&spld, $&sple, $&splf

```

$$\begin{aligned}
&\left[z - 3y - 5x + 4u + \frac{4}{j+1}, z - 3y - 5x + 4u + \frac{4}{j+2}, z - 3y - 5x + 4u + \frac{4}{j+3} \right] \\
&\left[6y + 3x - w + 3u + \frac{2}{j+1}, 6y + 3x - w + 3u + \frac{2}{j+2}, 6y + 3x - w + 3u + \frac{2}{j+3} \right] \\
&z + y + x + w + 6u = 1 \\
&\left[-z + 3y - 8x + 9w - \frac{3}{j+1}, -z + 3y - 8x + 9w - \frac{3}{j+2}, -z + 3y - 8x + 9w - \frac{3}{j+3} \right] \\
&\left[-3z + 6y - 3x + 5w + 2u + \frac{1}{j+1}, -3z + 6y - 3x + 5w + 2u + \frac{1}{j+2}, -3z + 6y - 3x + 5w + 2u + \frac{1}{j+3} \right] \\
&\left[5z + y - 2x + 3w + u + \frac{4}{j+1}, 5z + y - 2x + 3w + u + \frac{4}{j+2}, 5z + y - 2x + 3w + u + \frac{4}{j+3} \right]
\end{aligned}$$

Cari nilai-nilainya

```

>sspl &= solve([spla,splb,splc,spld,sple,splf],[u,v,w,x,y,z]);
$&sspl

```

□

unicodehyperref hyphensurl []book xcolor amsmath,amssymb
iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype
[protrusion]basicmath parskip graphicx bookmark xurl

MENGGAMBAR GRAFIK 2D DENGAN EMT

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

Plot Dasar

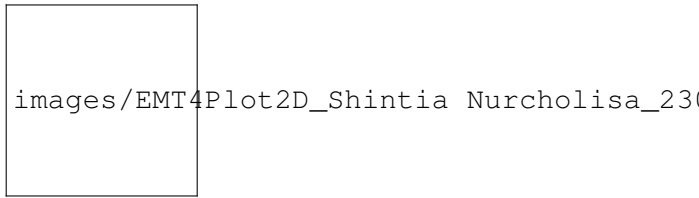
Ada fungsi plot yang sangat mendasar. Terdapat koordinat layar yang selalu berkisar antara 0 hingga 1024 di setiap sumbu, tidak peduli apakah layarnya berbentuk persegi atau tidak. Semut terdapat koordinat plot, yang dapat diatur dengan `setplot()`. Pemetaan antar koordinat bergantung pada jendela plot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

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

```
>clg; // 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
>reset;
```

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

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



Gambar 12.1 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-001.png

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

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

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

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



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.2 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-002.png

Aspek Plot

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

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

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



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.3 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-003.png

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

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

PLOT 2D DI EULER

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

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

Euler dapat membuat plot 2D

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

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

PLOT EKSPRESI ATAU VARIABEL

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

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

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

```
>plot2d(“x^2”):
```



Gambar 12.4 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-004.png

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



Gambar 12.5 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-005.png

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

Dari beberapa contoh sebelumnya Anda dapat melihat bahwa aslinya gambar plot menggunakan sumbu X dengan rentang nilai dari



Gambar 12.6 images/EMT4Plot2D_Shintia%20Nurcholisa.23030630071-006.png

-2 sampai dengan 2. Untuk mengubah rentang nilai X dan Y, Anda dapat menambahkan nilai-nilai batas X (dan Y) di belakang ekspresi yang digambar.

Rentang plot diatur dengan parameter yang ditetapkan sebagai berikut

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

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



Gambar 12.7 images/EMT4Plot2D_Shintia%20Nurcholisa.23030630071-007.png

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

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



Gambar 12.8 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-008.png



Gambar 12.9 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-009.png

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

Dalam opsi, plot dapat diatur agar muncul

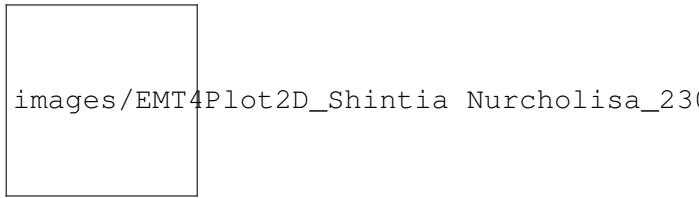
- di jendela terpisah yang dapat diubah ukurannya,
- di jendela buku catatan.

Lebih banyak gaya dapat dicapai dengan perintah plot tertentu.

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

Untuk membagi jendela menjadi beberapa plot, gunakan perintah `figure()`. Dalam contoh, kita memplot x^1 hingga x^4 menjadi 4 bagian jendela. gambar(0) mengatur ulang jendela default.

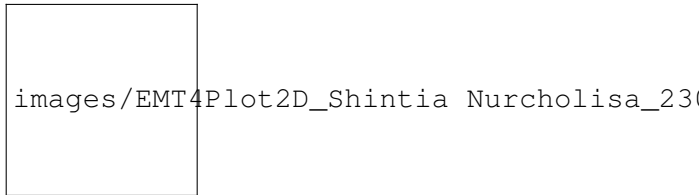
```
>reset;
>figure(2,2); ...
> for n=1 to 4; figure(n); plot2d("x^"+n); end; ...
> figure(0);
```



Gambar 12.10 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-010.png

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

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



Gambar 12.11 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-011.png

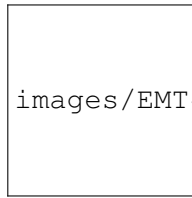
Jika argumen pada `plot2d()` adalah ekspresi yang diikuti oleh empat angka, angka-angka tersebut adalah rentang x dan y untuk plot tersebut.

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

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

```
>aspect(1.5); plot2d("sin(x)",0,2pi,-1.2,1.2,grid=3,xl="x",yl="sin(x)");
>plot2d("sin(x)+cos(2*x)",0,4pi):
```

Gambar yang dihasilkan dengan memasukkan plot ke dalam



Gambar 12.12 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-012.png



Gambar 12.13 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-013.png

jendela teks disimpan di direktori yang sama dengan buku catatan, secara default di subdirektori bernama “gambar”. Mereka juga digunakan oleh ekspor HTML.

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

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

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

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

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

```
>plot2d(“log(x)”,-0.1,2):
```

Parameter square=true (atau >square) memilih rentang y secara



Gambar 12.14 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-014.png



Gambar 12.15 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-015.png



Gambar 12.16 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-016.png

otomatis sehingga hasilnya adalah jendela plot persegi. Perhatikan bahwa secara default, Euler menggunakan spasi persegi di dalam jendela plot.

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



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.17 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-017.png

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



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.18 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-018.png

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

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

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

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

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

```
>plot2d(expr,-2,2): // plot from -2 to 2
```

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

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



Gambar 12.19 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-019.png



Gambar 12.20 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-020.png

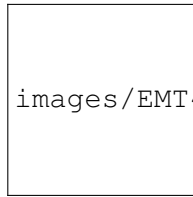


Gambar 12.21 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-021.png



Gambar 12.22 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-022.png

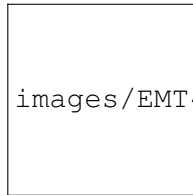
another plot



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.23 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-023.png

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



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.24 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-024.png

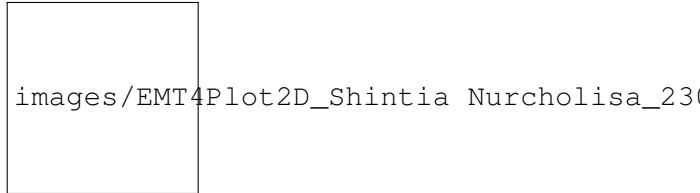
```
>plot2d(&diff(expr,x),a=-2,b=2,>square): // keep plot square
```



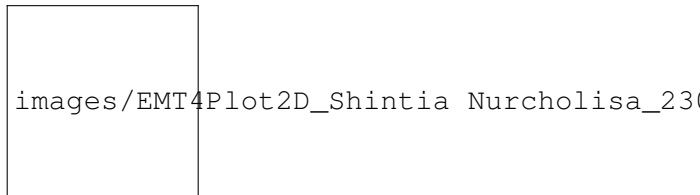
images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.25 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-025.png

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



Gambar 12.26 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-026.png



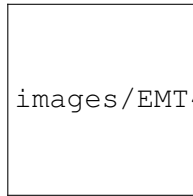
Gambar 12.27 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-027.png

FUNGSI DALAM SATU PARAMETER

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

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

```
>function f(x,a) := x2/a+a*x2-x; // define a function  
>a=0.3; plot2d(“f”,0,1;a) // plot with a=0.3
```



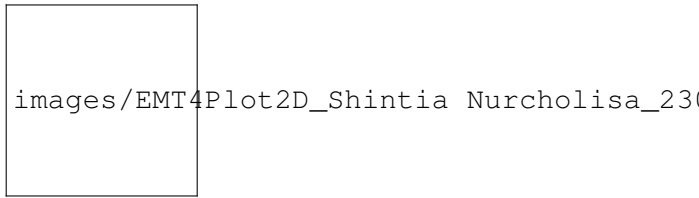
Gambar 12.28 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-028.png

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

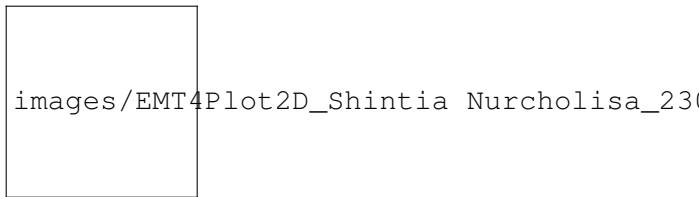


Gambar 12.29 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-029.png

```
>plot2d({{“f”,0.2}},0,1) // plot with a=0.2  
>plot2d({{“f(x,b)”,b=0.1}},0,1) // plot with 0.1
```

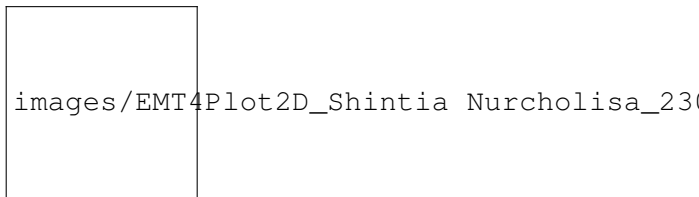


Gambar 12.30 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-030.png



Gambar 12.31 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-031.png

```
>function f(x) := x^3-x; ...  
> plot2d("f",r=1):
```



Gambar 12.32 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-032.png

Berikut ini ringkasan fungsi yang diterima

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

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

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

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

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



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.33 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-033.png

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

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

$$e^{-x} \sin(x)$$

```
>plot2d(expr,0,3pi):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.34 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-034.png

```
>function f(x) &= x^x;
```

```
>plot2d(f,r=1,cx=1,cy=1,color=blue,thickness=2);
```




images/EMT4Plot2D_Shintia Nurcholisa_23030630071-035.png

Gambar 12.35 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-035.png

```
>plot2d(&diff(f(x),x),>add,color=red,style="-.-"):
```

Untuk gaya garis ada berbagai pilihan.

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

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

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

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

Berikut adalah tampilan warna EMT yang telah ditentukan sebelumnya.

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

Tapi Anda bisa menggunakan warna apa saja.

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



Gambar 12.36 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-036.png



Gambar 12.37 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-037.png



Gambar 12.38 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-038.png

MENG GAMBAR BEBERAPA KURVA PADA BIDANG KOORDINAT YANG SAMA

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

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



Gambar 12.39 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-039.png

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

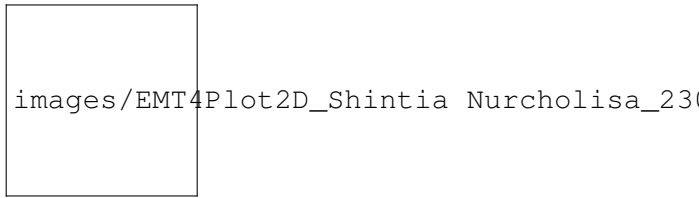


Gambar 12.40 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-040.png

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

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

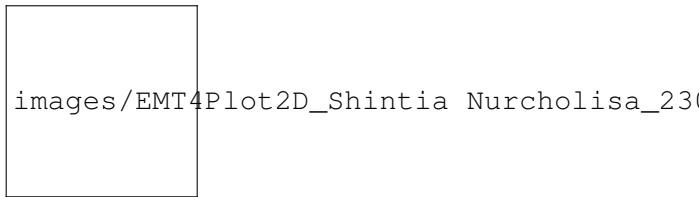
Kita tambahkan titik perpotongan dengan label (pada posisi "cl" untuk kiri tengah), dan masukkan hasilnya ke dalam buku catatan.



Gambar 12.41 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-041.png

Kami juga menambahkan judul pada plot.

```
> plot2d(["cos(x)", "x"], r=1.1, cx=0.5, cy=0.5, ...
> color=[black, blue], style=["-", ":"], ...
> grid=1);
> x0=solve("cos(x)-x", 1); ...
> plot2d(x0, x0, >points, >add, title="Intersection Demo"); ...
> label("cos(x) = x", x0, x0, pos="cl", offset=20):
```



Gambar 12.42 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-042.png

Dalam demo berikut, kita memplot fungsi $\sin(x)=\sin(x)/x$ dan ekspansi Taylor ke-8 dan ke-16. Kami menghitung perluasan ini menggunakan Maxima melalui ekspresi simbolik.

Plot ini dilakukan dalam perintah multi-baris berikut dengan tiga panggilan ke `plot2d()`. Yang kedua dan ketiga memiliki kumpulan tanda `>add`, yang membuat plot menggunakan rentang sebelumnya.

Kami menambahkan kotak label yang menjelaskan fungsinya.

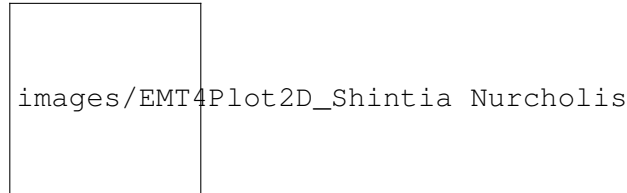
```
> $taylor(sin(x)/x, x, 0, 4)
```

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

```

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

```



Gambar 12.43 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-044.png

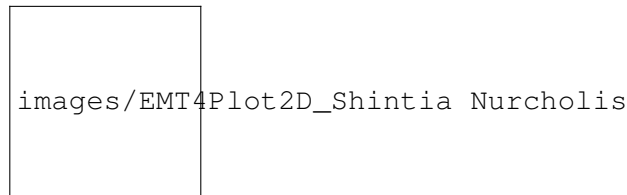
Dalam contoh berikut, kami menghasilkan Polinomial Bernstein.

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

```

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

```



Gambar 12.44 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-046.png

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

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

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



images/EMT4Plot2D_Shintia Nurcholisa_23030630071-047.png

Gambar 12.45 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-047.png

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

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



images/EMT4Plot2D_Shintia Nurcholisa_23030630071-048.png

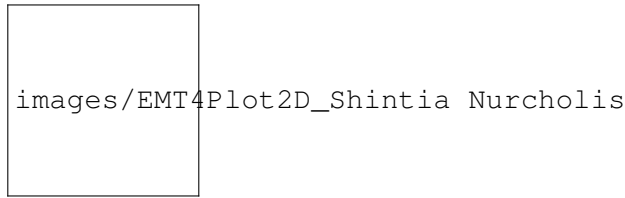
Gambar 12.46 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-048.png

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

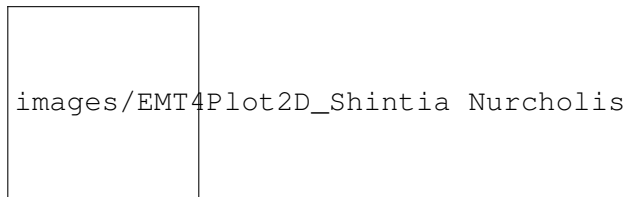
```
>plot2d(["sin(x)","cos(x)"],0,2pi,color=4:5):
>plot2d(["sin(x)","cos(x)"],0,2pi): // plot vector of expressions
```

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

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



Gambar 12.47 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-049.png



Gambar 12.48 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-050.png

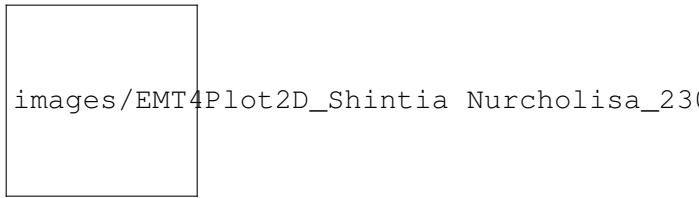
$$\begin{aligned} & \left[\frac{(1-x)^6}{6^4}, \frac{10(1-x)^5}{5^5}x, \frac{45(1-x)^4}{4^6}x^2, \frac{120(1-x)^3}{3^7}x^3, \frac{210(1-x)^2}{2^8}x^4, \frac{252(1-x)}{9^{10}}x^5, \frac{210}{10^{10}}x^6, \frac{45}{45^{10}}x^7, \frac{10}{10^{10}}x^8, x^9 \right] \end{aligned}$$

`>mxm2str(v) // get a vector of strings from the symbolic vector`

```
(1-x)^10
10*(1-x)^9*x
45*(1-x)^8*x^2
120*(1-x)^7*x^3
210*(1-x)^6*x^4
252*(1-x)^5*x^5
210*(1-x)^4*x^6
120*(1-x)^3*x^7
45*(1-x)^2*x^8
10*(1-x)*x^9
x^10
```



```
>plot2d(mxm2str(v),0,1): // plot functions
```



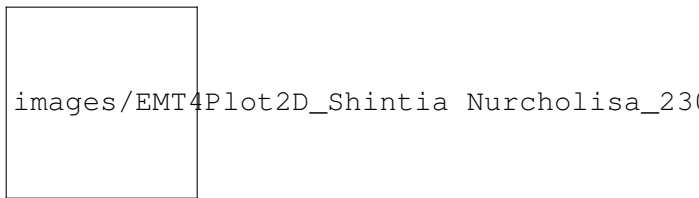
Gambar 12.49 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-051.png

Alternatif lain adalah dengan menggunakan bahasa matriks Euler.

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

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

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



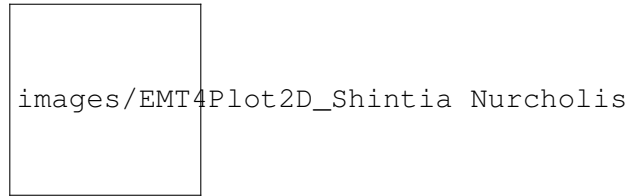
Gambar 12.50 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-052.png

Ekspresi dan fungsi satu baris dapat melihat variabel global.

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

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

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

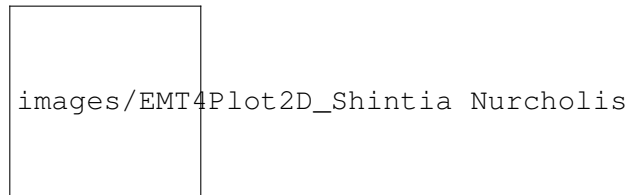


Gambar 12.51 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-053.png

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

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

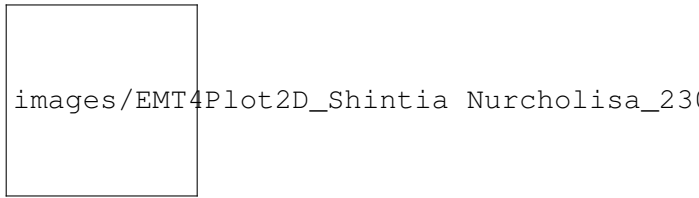
```
>plot2d({{"f",1}},-10,10); ...
> for a=2:10; plot2d({{"f",a}},>add); end:
```



Gambar 12.52 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-054.png

Kita dapat mencapai hasil yang sama dengan cara berikut menggunakan bahasa matriks EMT. Setiap baris matriks $f(x,a)$ merupakan satu fungsi. Selain itu, kita dapat mengatur warna untuk setiap baris matriks. Klik dua kali pada fungsi `getspectral()` untuk penjelasannya.

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



Gambar 12.53 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-055.png

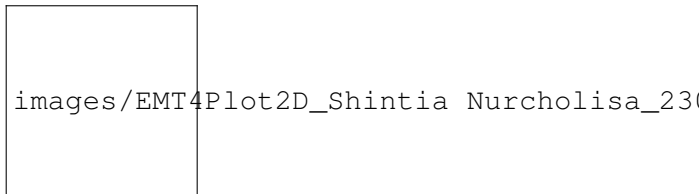
Label Teks

Dekorasi sederhana pun bisa

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

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

```
>plot2d(“x3-x”,-1,2,title=“y=x3-x”,yl=“y”,xl=“x”):
```



Gambar 12.54 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-056.png

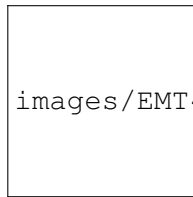
```
>expr := “log(x)/x”; ...
> plot2d(expr,0.5,5,title=“y=”+expr,xl=“x”,yl=“y”); ...
> label(“(1,0)”,1,0); label(“Max”,E,expr(E),pos=“lc”):
```

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



Gambar 12.55 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-057.png

```
>function f(x) &= x2*exp(-x2); ...  
> plot2d(&f(x),a=-3,b=3,c=-1,d=1); ...  
> plot2d(&diff(f(x),x),>add,color=blue,style="--"); ...  
> labelbox(["function","derivative"],styles=["-","-"], ...  
> colors=[black,blue],w=0.4):
```



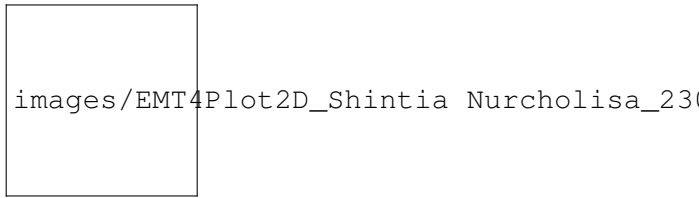
Gambar 12.56 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-058.png

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

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

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

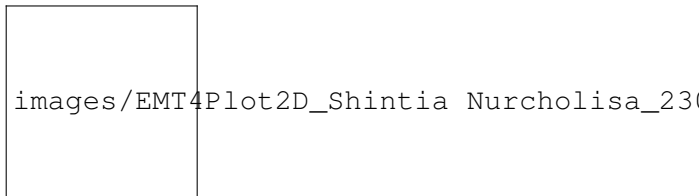
```
>n=10; plot2d(0:n,bin(n,0:n),>addpoints); ...  
> labelbox("Binomials",styles="[]",>points,x=0.1,y=0.1, ...  
> tcolor=black,>left):
```



Gambar 12.57 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-059.png

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

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



Gambar 12.58 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-060.png

Fitur serupa adalah fungsi `textbox()`.

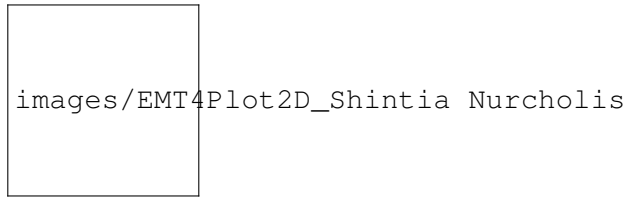
Lebar nya secara default adalah lebar maksimal baris teks. Tapi itu bisa diatur oleh pengguna juga.

```
>function f(x) &= exp(-x)*sin(2*pi*x); ...
> plot2d("f(x)",0,2pi); ...
> textbox(latex("\text{Example of a damped oscillation}\ f(x)=e^{-x}\sin(2\pi x)"),w=0.85):
```

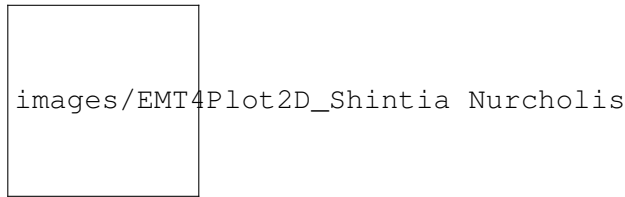
Label teks, judul, kotak label, dan teks lainnya dapat berisi string Unicode (lihat sintaks EMT untuk mengetahui lebih lanjut tentang string Unicode).

```
>plot2d("x^3-x",title="u"x → x3 - x");
```

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

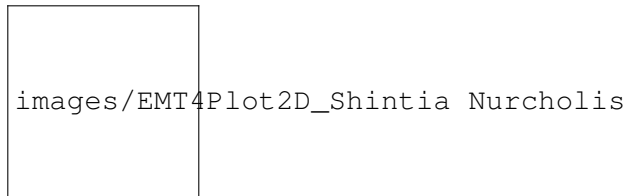


Gambar 12.59 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-061.png



Gambar 12.60 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-062.png

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



Gambar 12.61 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-063.png

LaTeX

Anda juga dapat memplot rumus LaTeX jika Anda telah menginstal sistem LaTeX. Saya merekomendasikan MiKTeX. Jalur ke biner “lateks” dan “dvi2png” harus berada di jalur sistem, atau Anda harus mengatur LaTeX di menu opsi.

Perhatikan, penguraian LaTeX lambat. Jika Anda ingin menggunakan LaTeX dalam plot animasi, Anda harus memanggil

latex() sebelum loop satu kali dan menggunakan hasilnya (gambar dalam matriks RGB).

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

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



Gambar 12.62 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-064.png

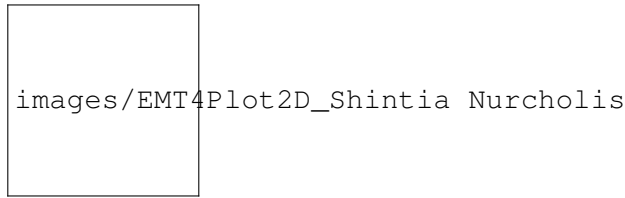
Seringkali, kita menginginkan spasi dan label teks yang tidak konformal pada sumbu x. Kita bisa menggunakan xaxis() dan yaxis() seperti yang akan kita tunjukkan nanti.

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

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

Tentu saja fungsinya juga bisa digunakan.

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



Gambar 12.63 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-065.png

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

Parameter “peta” membantu menggunakan fungsi untuk vektor. Untuk

plot, itu tidak perlu. Tapi untuk menunjukkan vektorisasi itu berguna, kita menambahkan beberapa poin penting ke plot di $x=-1$, $x=0$ dan $x=1$.

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

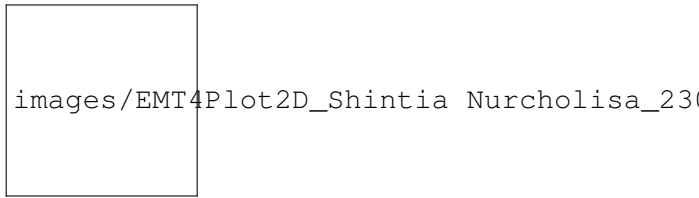
dua label dan kotak teks. Tentu saja, Anda hanya bisa menggunakannya

LaTeX jika Anda telah menginstal LaTeX dengan benar.

```
>plot2d(“f”,-1,1,xl=“x”,yl=“f(x)”,grid=6); ...
> plot2d([-1,0,1],f([-1,0,1]),>points,>add); ...
> label(latex(“x^3”),0.72,f(0.72)); ...
> label(latex(“x^2”),-0.52,f(-0.52),pos=“ll”); ...
> textbox( ...
> latex(“f(x)=\begin{cases} x^3 & x>0 \\ x^2 & x \le 0 \end{cases}”),
...
> x=0.7,y=0.2):
```

Interaksi Pengguna

Saat memplot suatu fungsi atau ekspresi, parameter `>pengguna` memungkinkan pengguna untuk memperbesar dan menggeser plot



Gambar 12.64 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-066.png

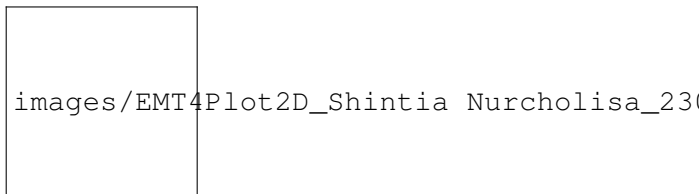
dengan tombol kursor atau mouse. Pengguna bisa

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

Tombol spasi akan mengatur ulang plot ke jendela plot aslinya.

Saat memplot data, flag `>user` hanya akan menunggu penekanan tombol.

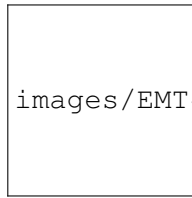
```
>plot2d({{"x^3-a*x",a=1}},>user,title="Press any key!"):
```



Gambar 12.65 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-067.png

```
>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).



Gambar 12.66 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-068.png

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

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

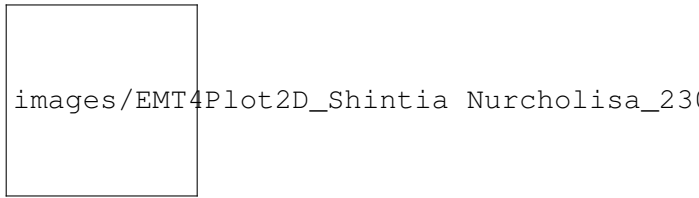
```
>function plotf(xp,yp,select) ...  
  
    d=interp(xp,yp);  
    plot2d("interpval(xp,d,x)";d,xp,r=2);  
    plot2d(xp,yp,>points,>add);  
    if select>0 then  
        plot2d(xp[select],yp[select],color=red,>points,>add);  
    endif;  
    title("Drag one point, or press space or return!");  
endfunction
```

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

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

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

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



Gambar 12.67 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-069.png

Pertama kita membutuhkan fungsi plot.

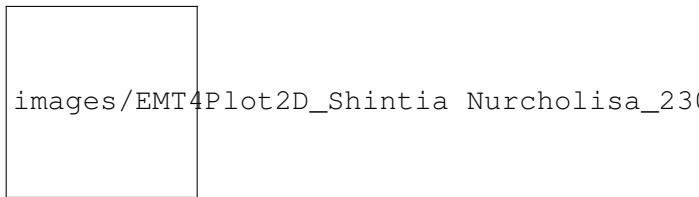
```
>function plotf([a,b]) := plot2d("exp(a*x)*cos(2pi*b*x)",0,2pi;a,b);
```

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

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

```
>dragvalues("plotf",["a","b],[-1,2],[[-2,2];[1,10]], ...
```

```
> heading="Drag these values:",hcolor=black):
```



Gambar 12.68 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-070.png

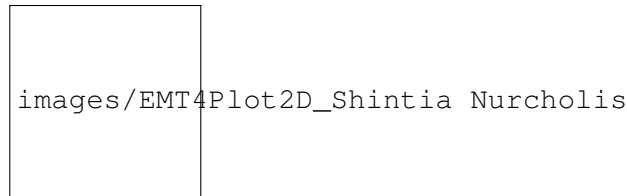
Dimungkinkan untuk membatasi nilai yang diseret menjadi bilangan bulat. Sebagai contoh, kita menulis fungsi plot, yang memplot polinomial Taylor berderajat n ke fungsi kosinus.

```
>function plotf(n) ...
```

```
plot2d("cos(x)",0,2pi,>square,grid=6);
plot2d("&taylor(cos(x),x,0,@n)",color=blue,>add);
textbox("Taylor polynomial of degree "+n,0.1,0.02,style="t
endfunction
```

Sekarang kita izinkan derajat n bervariasi dari 0 hingga 20 dalam 20 perhentian. Hasil `dragvalues()` digunakan untuk memplot sketsa dengan n ini, dan untuk memasukkan plot ke dalam buku catatan.

```
> nd=dragvalues("plotf","degree",2,[0,20],20,y=0.8, ...
> heading="Drag the value:"); ...
> plotf(nd):
```



Gambar 12.69 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-071.png

Berikut ini adalah demonstrasi sederhana dari fungsinya. Pengguna dapat menggambar jendela plot, meninggalkan jejak titik.

```
>function dragtest ...

plot2d(none,r=1,title="Drag with the mouse, or pres
start=0;
repeat
  {flag,m,time}=mousedrag();
  if flag==0 then return; endif;
  if flag==2 then
    hold on; mark(m[1],m[2]); hold off;
  endif;
end
endfunction
```

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

Gaya Plot 2D

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

dan judul dapat ditambahkan secara manual. Untuk menyetel ulang ke gaya default, gunakan `reset()`.

```
> aspect();  
> figure(3,4); ...  
> figure(1); plot2d("x^3-x",grid=0); ... // no grid, frame or axis  
    > figure(2); plot2d("x^3-x",grid=1); ... // x-y-axis  
    > figure(3); plot2d("x^3-x",grid=2); ... // default ticks  
    > figure(4); plot2d("x^3-x",grid=3); ... // x-y- axis with labels  
inside  
    > figure(5); plot2d("x^3-x",grid=4); ... // no ticks, only labels  
    > figure(6); plot2d("x^3-x",grid=5); ... // default, but no margin  
    > figure(7); plot2d("x^3-x",grid=6); ... // axes only  
    > figure(8); plot2d("x^3-x",grid=7); ... // axes only, ticks at  
axis  
    > figure(9); plot2d("x^3-x",grid=8); ... // axes only, finer ticks  
at axis  
    > figure(10); plot2d("x^3-x",grid=9); ... // default, small ticks  
inside  
    > figure(11); plot2d("x^3-x",grid=10); ...// no ticks, axes only  
    > figure(0);
```



images/EMT4Plot2D_Shintia Nurcholisa_23030630071-072.png

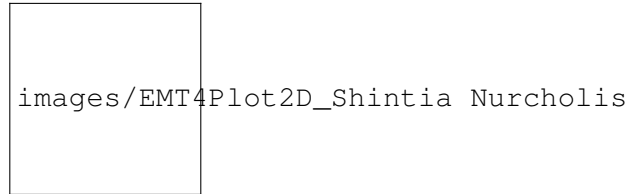
Gambar 12.70 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-072.png

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

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

```
> aspect(1.5);
```

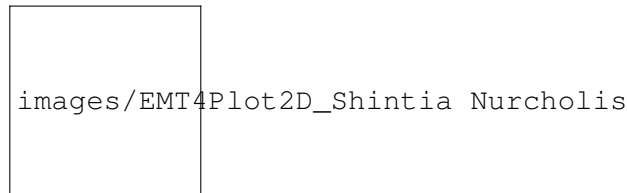
```
>plot2d("x^3-x",grid=0); // plot
>frame; xgrid([-1,0,1]); ygrid(0): // add frame and grid
```



Gambar 12.71 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-073.png

Untuk judul plot dan label sumbu, lihat contoh berikut.

```
>plot2d("exp(x)",-1,1);
>textcolor(black); // set the text color to black
>title(latex("y=e^x")); // title above the plot
>xlabel(latex("x")); // "x" for x-axis
>ylabel(latex("y"),>vertical); // vertical "y" for y-axis
>label(latex("(0,1)"),0,1,color=blue): // label a point
```



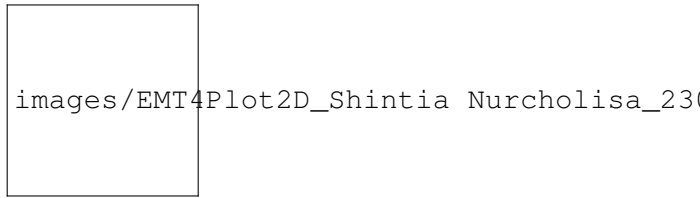
Gambar 12.72 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-074.png

Sumbu dapat digambar secara terpisah dengan `xaxis()` dan `yaxis()`.

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

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

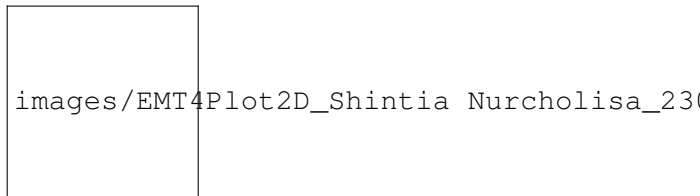
```
>function f(x) &= x^3-x
```



Gambar 12.73 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-075.png

$$x^3 - x$$

```
>plot2d(f,-1,1,>square);
>x0=fmin(f,0,1); // compute point of minimum
>label("Rel. Min.",x0,f(x0),pos="lc"); // add a label there
```



Gambar 12.74 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-076.png

Ada juga kotak teks.

```
>plot2d(&f(x),-1,1,-2,2); // function
>plot2d(&diff(f(x),x),>add,style="-",color=red); // derivative
>labelbox(["f","f'"],["-","-"],[black,red]); // label box
>plot2d(["exp(x)","1+x"],color=[black,blue],style=["-","-."]);
>gridstyle("-",color=gray,textcolor=gray,framecolor=gray);
...
> plot2d("x^3-x",grid=1); ...
> settitle("y=x^3-x",color=black); ...
> label("x",2,0,pos="bc",color=gray); ...
> label("y",0,6,pos="cl",color=gray); ...
> reset();
```



Gambar 12.75 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-077.png



Gambar 12.76 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-078.png



Gambar 12.77 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-079.png

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

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

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



images/EMT4Plot2D_Shintia Nurcholisa_23030630071-080.png

Gambar 12.78 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-080.png

Berikut adalah contoh lain, di mana string Unicode digunakan dan sumbunya berada di luar area plot.

```
> aspect(1.5);
> plot2d(["sin(x)","cos(x)"],0,2pi,color=[red,green],<grid,<frame);
...
> xaxis(-1.1,(0:2)*pi,xt=["0","u","u^2"],style="-",>ticks,>zero); ...
> xgrid((0:0.5:2)*pi,<ticks); ...
> yaxis(-0.1*pi,-1:0.2:1,style="-",>zero,>grid); ...
> labelbox(["sin","cos"],colors=[red,green],x=0.5,y=0.2,>left); ...
> xlabel(u""); ylabel(u"f()");
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.79 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-081.png

PLOTTING 2D DATA

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

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

Tapi Anda bisa memasukkan data secara langsung.

- Gunakan vektor baris untuk x dan y untuk satu fungsi.
- Matriks untuk x dan y diplot baris demi baris.

Berikut adalah contoh dengan satu baris untuk x dan y .

```
>x=-10:0.1:10; y=exp(-x^2)*x; plot2d(x,y):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.80 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-082.png

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

- `style="..."`: Pilih dari `[]`, `"<>"`, `"o"`, `"."`, `".."`, `"+"`, `"*"`, `"[]#"`, `"<#"`, `"o#"`, `"..#"`, `"#"`, `"|"`.

Untuk memplot kumpulan titik, gunakan `>titik`. Jika warna merupakan vektor warna, masing-masing titik

mendapat warna berbeda. Untuk matriks koordinat dan vektor kolom, warna diterapkan pada baris matriks.

Parameter `>addpoints` menambahkan titik ke segmen garis untuk plot data.

```
>xdata=[1,1.5,2.5,3,4]; ydata=[3,3.1,2.8,2.9,2.7]; // data  
>plot2d(xdata,ydata,a=0.5,b=4.5,c=2.5,d=3.5,style=":"); //  
lines  
>plot2d(xdata,ydata,>points,>add,style="o"): // add points
```



Gambar 12.81 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-083.png

```
>p=polyfit(xdata,ydata,1); // get regression line  
>plot2d("polyval(p,x)",>add,color=red): // add plot of line
```



Gambar 12.82 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-084.png

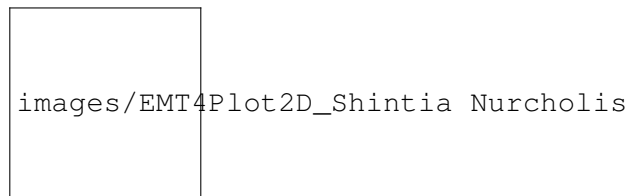
MENG GAMBAR DAERAH YANG DIBATASI KURVA

Plot data sebenarnya berbentuk poligon. Kita juga dapat memplot kurva atau kurva terisi.

- terisi=benar mengisi plot.
- style="...": Pilih dari "#", "/", ",", "/".
- Fillcolor : Lihat di atas untuk mengetahui warna yang tersedia.

Warna isian ditentukan oleh argumen "fillcolor", dan pada <outline opsional, mencegah menggambar batas untuk semua gaya kecuali gaya default.

```
>t=linspace(0,2pi,1000); // parameter for curve
>x=sin(t)*exp(t/pi); y=cos(t)*exp(t/pi); // x(t) and y(t)
>figure(1,2); aspect(16/9)
>figure(1); plot2d(x,y,r=10); // plot curve
>figure(2); plot2d(x,y,r=10,>filled,style="/",fillcolor=red); //
fill curve
>figure(0):
```



Gambar 12.83 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-085.png

Dalam contoh berikut kita memplot elips terisi dan dua segi enam terisi menggunakan kurva tertutup dengan 6 titik dengan gaya isian berbeda.

```
>x=linspace(0,2pi,1000); plot2d(sin(x),cos(x)*0.5,r=1,>filled,style="/"):
```



Gambar 12.84 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-086.png

```
>t=linspace(0,2pi,6); ...  
> plot2d(cos(t),sin(t),>filled,style="/",fillcolor=red,r=1.2):
```



Gambar 12.85 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-087.png

```
>t=linspace(0,2pi,6); plot2d(cos(t),sin(t),>filled,style="#"):
```



Gambar 12.86 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-088.png

Contoh lainnya adalah septagon yang kita buat dengan 7 titik pada lingkaran satuan.

```
>t=linspace(0,2pi,7); ...  
> plot2d(cos(t),sin(t),r=1,>filled,style="/",fillcolor=red):
```

Berikut adalah himpunan nilai maksimal dari empat kondisi



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.87 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-089.png

linier yang kurang dari atau sama dengan 3. Ini adalah $A[k].v \leq 3$ untuk semua baris A. Untuk mendapatkan sudut yang bagus, kita menggunakan n yang relatif besar.

```
>A=[2,1;1,2;-1,0;0,-1];
>function f(x,y) := max([x,y].A');
>plot2d("f",r=4,level=[0;3],color=green,n=111):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.88 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-090.png

Poin utama dari bahasa matriks adalah memungkinkan pembuatan tabel fungsi dengan mudah.

```
>t=linspace(0,2pi,1000); x=cos(3*t); y=sin(4*t);
```

Kami sekarang memiliki nilai vektor x dan y. plot2d() dapat memplot nilai-nilai ini sebagai kurva yang menghubungkan titik-titik tersebut. Plotnya bisa diisi. Dalam hal ini menghasilkan hasil yang bagus karena aturan belitan, yang digunakan untuk isi.

```
>plot2d(x,y,<grid,<frame,>filled):
```

Vektor interval diplot terhadap nilai x sebagai wilayah terisi antara nilai interval yang lebih rendah dan lebih tinggi.

Hal ini dapat berguna untuk memplot kesalahan perhitungan.



images/EMT4Plot2D_Shintia Nurcholisa_23030630071-091.png

Gambar 12.89 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-091.png

Tapi itu bisa juga dapat digunakan untuk memplot kesalahan statistik.

```
>t=0:0.1:1; ...
> plot2d(t,interval(t-random(size(t)),t+random(size(t))),style="|");
...
> plot2d(t,t,add=true):
```



images/EMT4Plot2D_Shintia Nurcholisa_23030630071-092.png

Gambar 12.90 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-092.png

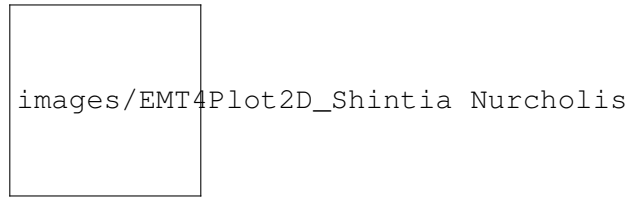
Jika x adalah vektor yang diurutkan, dan y adalah vektor interval, maka `plot2d` akan memplot rentang interval yang terisi pada bidang. Gaya isiannya sama dengan gaya poligon.

```
>t=-1:0.01:1; x=t-0.01,t+0.01; y=x^3-x;
>plot2d(t,y):
```

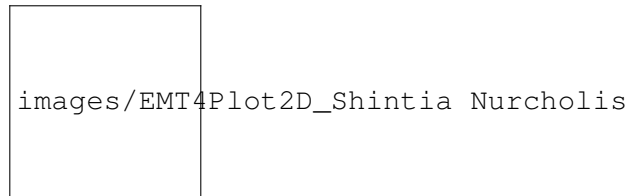
Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks $2 \times n$. Baris pertama adalah batas bawah dan baris kedua berisi batas atas.

```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y)
<= 1
```

Kita juga dapat mengisi rentang nilai seperti

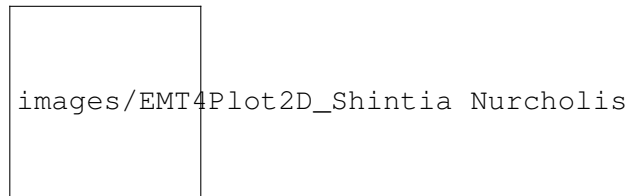


Gambar 12.91 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-093.png



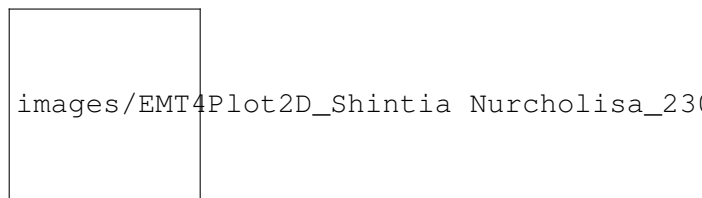
Gambar 12.92 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-094.png

```
>plot2d("(x2+y2)2-x2+y2",r=1.2,level=[-1;0],style="/"):
```



Gambar 12.93 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-095.png

```
>plot2d("cos(x)","sin(x)^3",xmin=0,xmax=2pi,>filled,style="/"):
```



Gambar 12.94 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-096.png

GRAFIK FUNGSI PARAMETRIK

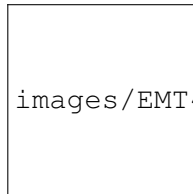
Nilai x tidak perlu diurutkan. (x,y) hanya menggambarkan sebuah kurva. Jika x diurutkan, kurva tersebut merupakan grafik suatu fungsi.

Dalam contoh berikut, kita memplot spiral

$$\gamma(t) = t \cdot (\cos(2\pi t), \sin(2\pi t))$$

Kita perlu menggunakan banyak titik untuk tampilan yang halus atau fungsi `adaptive()` untuk mengevaluasi ekspresi (lihat fungsi `adaptive()` untuk lebih jelasnya).

```
>t=linspace(0,1,1000); ...  
> plot2d(t*cos(2*pi*t),t*sin(2*pi*t),r=1):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.95 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-098.png

Sebagai alternatif, dimungkinkan untuk menggunakan dua ekspresi untuk kurva. Berikut ini plot kurva yang sama seperti di atas.

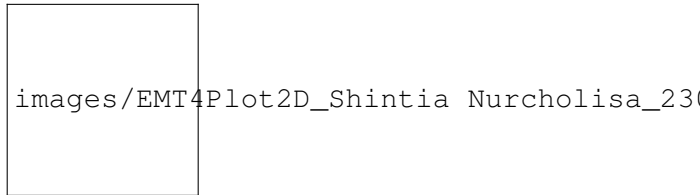
```
>plot2d("x*cos(2*pi*x)","x*sin(2*pi*x)",xmin=0,xmax=1,r=1):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.96 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-099.png

```
>t=linspace(0,1,1000); r=exp(-t); x=r*cos(2pi*t); y=r*sin(2pi*t);
>plot2d(x,y,r=1):
```



Gambar 12.97 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-100.png

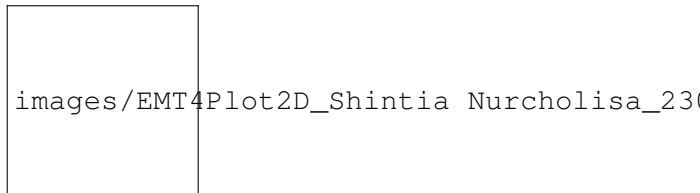
Pada contoh berikutnya, kita memplot kurvanya

$$\gamma(t) = (r(t) \cos(t), r(t) \sin(t))$$

dengan

$$r(t) = 1 + \frac{\sin(3t)}{2}.$$

```
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t); ...
> plot2d(x,y,>filled,fillcolor=red,style="l",r=1.5):
```



Gambar 12.98 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-103.png

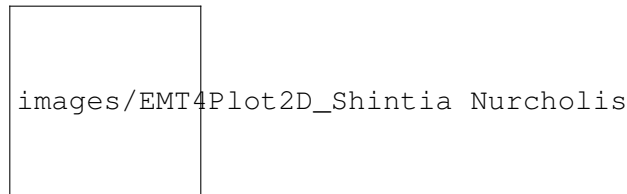
MENGGAMBAR GRAFIK BILANGAN KOMPLEKS

Serangkaian bilangan kompleks juga dapat diplot. Kemudian titik-titik grid akan dihubungkan. Jika sejumlah garis kisi ditentukan (atau vektor garis kisi 1×2) dalam argumen `cgrid`, hanya garis kisi tersebut yang terlihat.

Matriks bilangan kompleks secara otomatis akan diplot sebagai kisi-kisi pada bidang kompleks.

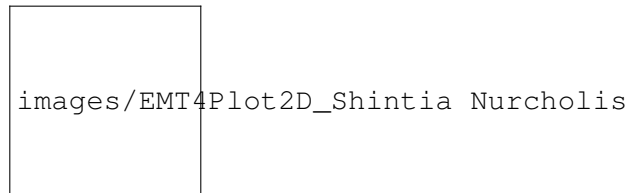
Pada contoh berikut, kita memplot gambar lingkaran satuan di bawah fungsi eksponensial. Parameter `cgrid` menyembunyikan beberapa kurva grid.

```
> aspect(); r=linspace(0,1,50); a=linspace(0,2pi,80)'; z=r*exp(I*a); ...  
> plot2d(z,a=-1.25,b=1.25,c=-1.25,d=1.25,cgrid=10):
```



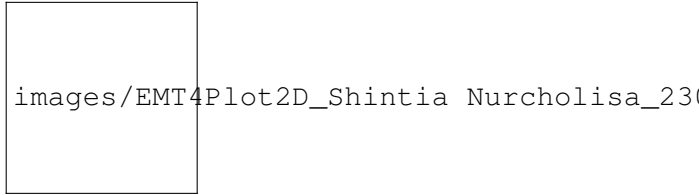
Gambar 12.99 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-104.png

```
> aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)';  
z=r*exp(I*a);  
> plot2d(exp(z),cgrid=[40,10]):
```



Gambar 12.100 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-105.png

```
>r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=r*exp(I*a);
>plot2d(exp(z),>points,>add):
```



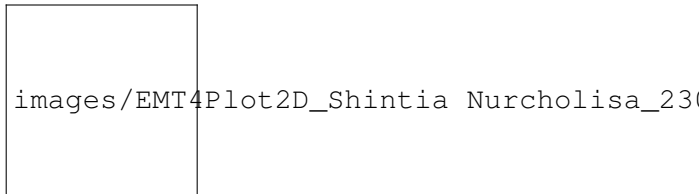
Gambar 12.101 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-106.png

Vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian nyata dan bagian imajiner.

Dalam contoh, kita memplot lingkaran satuan dengan

$$\gamma(t) = e^{it}$$

```
>t=linspace(0,2pi,1000); ...
> plot2d(exp(I*t)+exp(4*I*t),r=2):
```



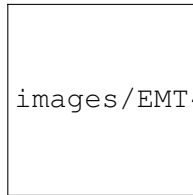
Gambar 12.102 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-108.png

PLOT STATISTIK

Ada banyak fungsi yang dikhususkan pada plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

Jumlah kumulatif dari nilai terdistribusi normal 0-1 menghasilkan jalan acak.

```
>plot2d(cumsum(randnormal(1,1000))):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.103 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-109.png

Penggunaan dua baris menunjukkan jalan dalam dua dimensi.

```
>X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.104 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-110.png

```
>columnsplot(cumsum(random(10)),style="l",color=blue):
```

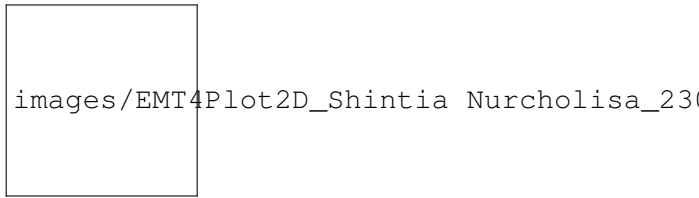
Itu juga dapat menampilkan string sebagai label.

```
>months=["Jan","Feb","Mar","Apr","May","Jun", ...
```

```
> "Jul","Aug","Sep","Oct","Nov","Dec"];
```

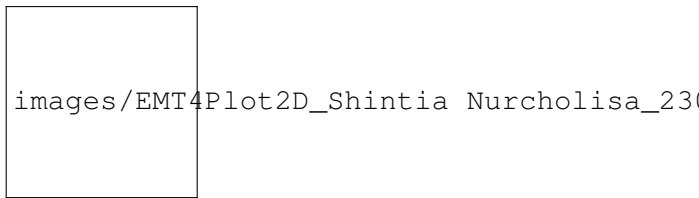
```
>values=[10,12,12,18,22,28,30,26,22,18,12,8];
```

```
>columnsplot(values,lab=months,color=red,style="-");
```

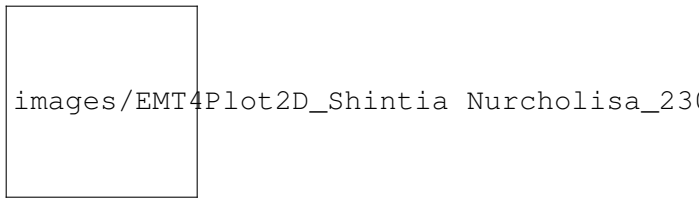
Gambar 12.105 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-111.png

```
>title("Temperature");
```



Gambar 12.106 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-112.png

```
>k=0:10;
>plot2d(k,bin(10,k),>bar):
```



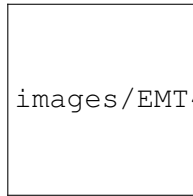
Gambar 12.107 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-113.png

```
>plot2d(k,bin(10,k)); plot2d(k,bin(10,k),>points,>add):
>plot2d(normal(1000),normal(1000),>points,grid=6,style="."):
>plot2d(normal(1,1000),>distribution,style="O"):
>plot2d("qnormal",0,5;2.5,0.5,>filled):
```

Untuk memplot distribusi statistik eksperimental, Anda dapat



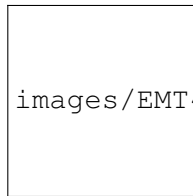
Gambar 12.108 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-114.png



Gambar 12.109 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-115.png



Gambar 12.110 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-116.png



Gambar 12.111 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-117.png

menggunakan `distribution=n` dengan `plot2d`.

```
>w=randexponential(1,1000); // exponential distribution  
>plot2d(w,>distribution): // or distribution=n with n intervals
```



images/EMT4Plot2D_Shintia Nurcholisa_23

Gambar 12.112 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-118.png

Atau Anda dapat menghitung distribusi dari data dan memplot hasilnya dengan `>bar` di `plot3d`, atau dengan `plot kolom`.

```
>w=normal(1000); // 0-1-normal distribution  
>{x,y}=histo(w,10,v=[-6,-4,-2,-1,0,1,2,4,6]); // interval bounds
```

v

```
>plot2d(x,y,>bar):
```



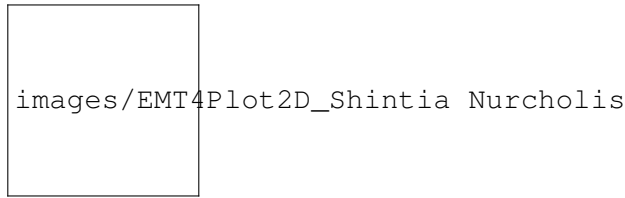
images/EMT4Plot2D_Shintia Nurcholisa_23

Gambar 12.113 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-119.png

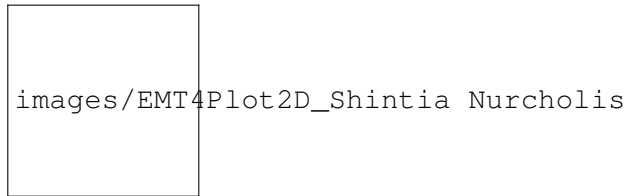
Fungsi `statplot()` mengatur gaya dengan string sederhana.

```
>statplot(1:10,cumsum(random(10)),"b");  
>n=10; i=0:n; ...  
> plot2d(i,bin(n,i)/2^n,a=0,b=10,c=0,d=0.3); ...  
> plot2d(i,bin(n,i)/2^n,points=true,style="ow",add=true,color=blue):
```

Selain itu, data dapat diplot sebagai batang. Dalam hal ini, `x` harus diurutkan dan satu elemen lebih panjang dari `y`. Batangnya akan memanjang dari `x[i]` hingga `x[i+1]` dengan nilai `y[i]`. Jika `x` berukuran



Gambar 12.114 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-120.png



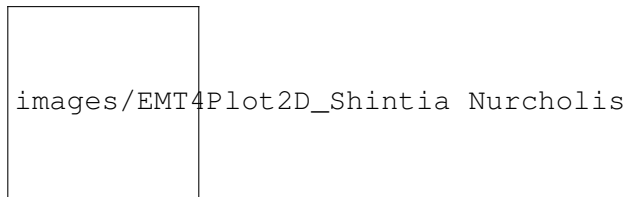
Gambar 12.115 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-121.png

sama dengan y, maka x akan diperpanjang satu elemen dengan spasi terakhir.

Gaya isian dapat digunakan seperti di atas.

```
>n=10; k=bin(n,0:n); ...
```

```
> plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray):
```



Gambar 12.116 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-122.png

Data untuk plot batang (batang=1) dan histogram (histogram=1) dapat diberikan secara eksplisit dalam xv dan yv, atau dapat dihitung dari distribusi empiris dalam xv dengan `>distribusi` (atau `distribusi=n`). Histogram nilai xv akan dihitung secara otomatis dengan `>histogram`.

Jika >even ditentukan, nilai xv akan dihitung dalam interval bilangan bulat.

```
>plot2d(normal(10000),distribution=50):
```



images/EMT4Plot2D_Shintia Nurcholisa_23

Gambar 12.117 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-123.png

```
>k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m,>bar):
```



images/EMT4Plot2D_Shintia Nurcholisa_23

Gambar 12.118 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-124.png

```
>columnplot(m,k):
```



images/EMT4Plot2D_Shintia Nurcholisa_23

Gambar 12.119 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-125.png

```
>plot2d(random(600)*6,histogram=6):
```

Untuk distribusi, terdapat parameter `distribution=n`, yang menghitung nilai secara otomatis dan mencetak distribusi relatif



Gambar 12.120 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-126.png

dengan n sub-interval.

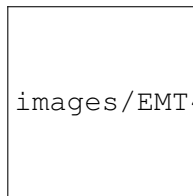
```
>plot2d(normal(1,1000),distribution=10,style="\"):
```



Gambar 12.121 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-127.png

Dengan parameter `even=true`, ini akan menggunakan interval bilangan bulat.

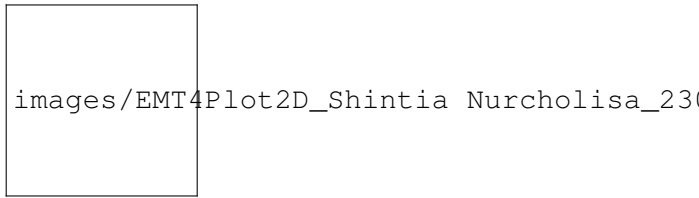
```
>plot2d(intrandom(1,1000,10),distribution=10,even=true):
```



Gambar 12.122 images/EMT4Plot2D_Shintia%20Nurcholis_23030630071-128.png

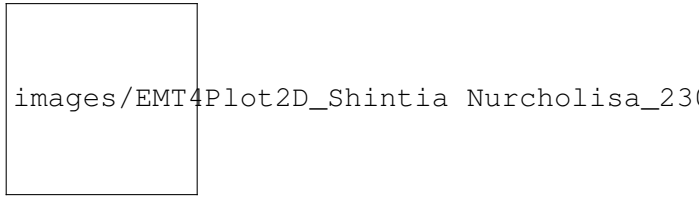
Perhatikan bahwa ada banyak plot statistik yang mungkin berguna. Silahkan lihat tutorial tentang statistik.

```
>columnsplot(getmultiplicities(1:6,intrandom(1,6000,6))):
```



Gambar 12.123 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-129.png

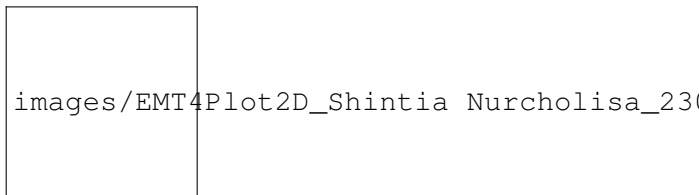
```
>plot2d(normal(1,1000),>distribution); ...  
> plot2d("qnormal(x)",color=red,thickness=2,>add):
```



Gambar 12.124 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-130.png

Ada juga banyak plot khusus untuk statistik. Plot kotak menunjukkan kuartil distribusi ini dan banyak outlier. Menurut definisinya, outlier dalam plot kotak adalah data yang melebihi 1,5 kali rentang 50% tengah plot.

```
>M=normal(5,1000); boxplot(quartiles(M)):
```



Gambar 12.125 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-131.png

FUNGSI IMPLISIT

Plot implisit menunjukkan penyelesaian garis level $f(x,y)=\text{level}$, dengan “level” dapat berupa nilai tunggal atau vektor nilai. Jika level = “auto”, akan ada garis level n_c , yang akan tersebar antara fungsi minimum dan maksimum secara merata. Warna yang lebih gelap atau terang dapat ditambahkan dengan $>\text{hue}$ untuk menunjukkan nilai fungsi. Untuk fungsi implisit, xv harus berupa fungsi atau ekspresi parameter x dan y , atau alternatifnya, xv dapat berupa matriks nilai.

Euler dapat menandai garis level

$$f(x, y) = c$$

dari fungsi apa pun.

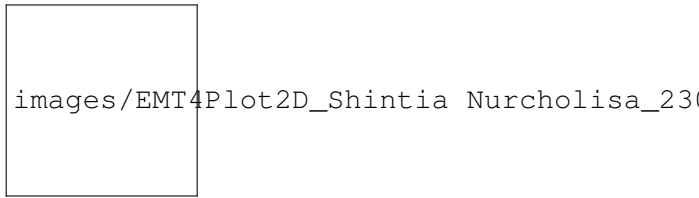
Untuk menggambar himpunan $f(x,y)=c$ untuk satu atau lebih konstanta c , Anda dapat menggunakan `plot2d()` dengan plot implisitnya pada bidang. Parameter c adalah `level=c`, dimana c dapat berupa vektor garis level. Selain itu, skema warna dapat digambar di latar belakang untuk menunjukkan nilai fungsi setiap titik dalam plot. Parameter “ n ” menentukan kehalusan plot.

```
>aspect(1.5);  
>plot2d(“x2+y2-x*y-x”,r=1.5,level=0,contourcolor=red):
```



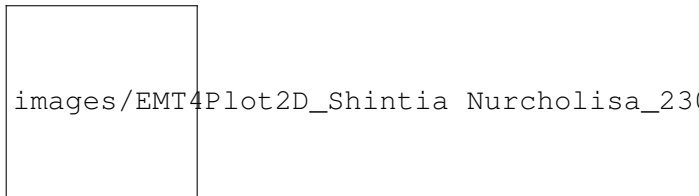
Gambar 12.126 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-133.png

```
>expr := “2*x2+x*y+3*y4+y”; // define an expression f(x,y)  
>plot2d(expr,level=0): // Solutions of f(x,y)=0  
>plot2d(expr,level=0:0.5:20,>hue,contourcolor=white,n=200):
```

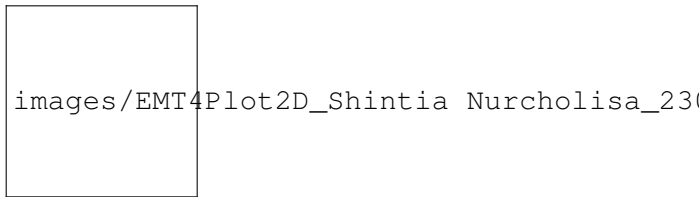
Gambar 12.127 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-134.png

// nice



Gambar 12.128 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-135.png

```
>plot2d(expr,level=0:0.5:20,>hue,>spectral,n=200,grid=4): //
nicer
```



Gambar 12.129 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-136.png

Ini juga berfungsi untuk plot data. Namun Anda harus menentukan rentangnya untuk label sumbu.

```
>x=-2:0.05:1; y=x'; z=expr(x,y);
>plot2d(z,level=0,a=-1,b=2,c=-2,d=1,>hue):
>plot2d("x3-y2",>contour,>hue,>spectral):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.130 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-137.png



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.131 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-138.png

```
>plot2d("x3-y2",level=0,contourwidth=3,>add,contourcolor=red):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.132 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-139.png

```
>z=z+normal(size(z))*0.2;
>plot2d(z,level=0.5,a=-1,b=2,c=-2,d=1):
>plot2d(expr,level=[0:0.2:5;0.05:0.2:5.05],color=lightgray):
>plot2d("x2+y3+x*y",level=1,r=4,n=100):
>plot2d("x2+2*y2-x*y",level=0:0.1:10,n=100,contourcolor=white,>hue):
```

Dimungkinkan juga untuk mengisi set



Gambar 12.133 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-140.png



Gambar 12.134 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-141.png



Gambar 12.135 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-142.png



Gambar 12.136 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-143.png

$$a \leq f(x, y) \leq b$$

dengan rentang level.

Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks 2xn. Baris pertama adalah batas bawah dan baris kedua berisi batas atas.

```
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y)
<= 1
```



Gambar 12.137 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-145.png

Plot implisit juga dapat menunjukkan rentang level. Maka level harus berupa matriks interval level 2xn, di mana baris pertama berisi awal dan baris kedua berisi akhir setiap interval. Alternatifnya, vektor baris sederhana dapat digunakan untuk level, dan parameter dl memperluas nilai level ke interval.

```
>plot2d("x4+y4",r=1.5,level=[0;1],color=blue,style="f"): 
```



Gambar 12.138 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-146.png

```
>plot2d("x2+y3+x*y",level=[0,2,4;1,3,5],style="f",r=2,n=100):
>plot2d("x2+y3+x*y",level=-10:20,r=2,style="-",dl=0.1,n=100):
>plot2d("sin(x)*cos(y)",r=pi,>hue,>levels,n=100):
```



Gambar 12.139 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-147.png



Gambar 12.140 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-148.png



Gambar 12.141 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-149.png

Dimungkinkan juga untuk menandai suatu wilayah

$$a \leq f(x, y) \leq b.$$

Hal ini dilakukan dengan menambahkan level dengan dua baris.

```
>plot2d("(x2+y2-1)3-x2*y3",r=1.3, ...  
> style="#",color=red,<outline, ...  
> level=[-2;0],n=100):
```



Gambar 12.142 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-151.png

Dimungkinkan untuk menentukan level tertentu. Misalnya, kita dapat memplot solusi persamaan seperti

$$x^3 - xy + x^2y^2 = 6$$

```
>plot2d("x3-x*y+x2*y2",r=6,level=1,n=100):
```



Gambar 12.143 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-153.png

```
>function starplot1 (v, style="f", color=green, lab=none) ...
```

```
if !holding() then clg; endif;  
w=window(); window(0,0,1024,1024);  
h=holding(1);
```

```

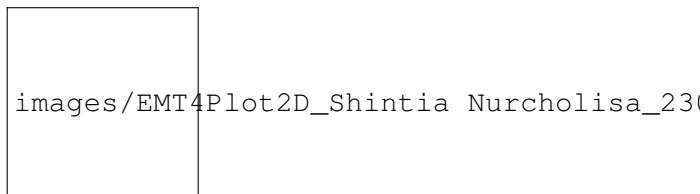
r=max(abs(v))*1.2;
setplot(-r,r,-r,r);
n=cols(v); t=linspace(0,2pi,n);
v=v|v[1]; c=v*cos(t); s=v*sin(t);
cl=barcolor(color); st=barstyle(style);
loop 1 to n
    polygon([0,c[#],c[#+1]], [0,s[#],s[#+1]],1);
    if lab!=none then
        rlab=v[#]+r*0.1;
        {col,row}=toscreen(cos(t[#])*rlab,sin(t[#])*rlab);
        ctext(""+lab[#],col,row-textheight()/2);
    endif;
end;
barcolor(cl); barstyle(st);
holding(h);
window(w);
endfunction

```

Tidak ada tanda centang kotak atau sumbu di sini. Selain itu, kami menggunakan jendela penuh untuk plotnya.

Kami memanggil reset sebelum kami menguji plot ini untuk mengembalikan default grafis. Ini tidak perlu dilakukan jika Anda yakin plot Anda berhasil.

```
>reset; starplot1(normal(1,10)+5,color=red,lab=1:10):
```



Gambar 12.144 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-154.png

Terkadang, Anda mungkin ingin merencanakan sesuatu yang plot2d tidak bisa lakukan, tapi hampir.

Dalam fungsi berikut, kita membuat plot impuls logaritmik. plot2d dapat melakukan plot logaritmik, tetapi tidak untuk batang

impuls.

```
>function logimpulseplot1 (x,y) ...  
  
    {x0,y0}=makeimpulse (x, log (y) /log (10) ) ;  
    plot2d(x0,y0,>bar,grid=0) ;  
    h=holding(1) ;  
    frame() ;  
    xgrid(ticks(x)) ;  
    p=plot() ;  
    for i=-10 to 10;  
        if i<=p[4] and i>=p[3] then  
            ygrid(i,yt="10^"+i) ;  
        endif;  
    end;  
    holding(h) ;  
endfunction
```

Mari kita uji dengan nilai yang terdistribusi secara eksponensial.

```
>aspect(1.5); x=1:10; y=-log(random(size(x)))*200; ...  
> logimpulseplot1(x,y):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.145 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-155.png

Mari kita menganimasikan kurva 2D menggunakan plot langsung. Perintah `plot(x,y)` hanya memplot kurva ke dalam jendela plot. `setplot(a,b,c,d)` menyatel jendela ini.

Fungsi `wait(0)` memaksa plot muncul di jendela grafis. Jika tidak, pengundian ulang akan dilakukan dalam interval waktu yang jarang.

```
>function animliss (n,m) ...
```



```

t=linspace(0,2pi,500);
f=0;
c=framecolor(0);
l=linewidth(2);
setplot(-1,1,-1,1);
repeat
    clg;
    plot(sin(n*t),cos(m*t+f));
    wait(0);
    if testkey() then break; endif;
    f=f+0.02;
end;
framecolor(c);
linewidth(l);
endfunction

```

Tekan tombol apa saja untuk menghentikan animasi ini.

>animliss(2,3); // lihat hasilnya, jika sudah puas, tekan ENTER

PLOT LOGARITMIK

EMT menggunakan parameter “logplot” untuk skala logaritmik.

Plot logaritma dapat diplot menggunakan skala logaritma di y dengan logplot=1, atau menggunakan skala logaritma di x dan y dengan logplot=2, atau di x dengan logplot=3.

- logplot=1: y-logaritma
- logplot=2: x-y-logaritma
- logplot=3: x-logaritma

```
>plot2d(“exp(x3-x)*x2”,1,5,logplot=1):
```



Gambar 12.146 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-156.png

```
>plot2d(“exp(x+sin(x))”,0,100,logplot=1):
```



Gambar 12.147 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-157.png

```
>plot2d(“exp(x+sin(x))”,10,100,logplot=2):
```



Gambar 12.148 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-158.png



Gambar 12.149 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-159.png

```
>plot2d("gamma(x)",1,10,logplot=1):  
>plot2d("log(x*(2+sin(x/100)))",10,1000,logplot=3):
```



Gambar 12.150 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-160.png

Ini juga berfungsi dengan plot data.

```
>x=10^(1:20); y=x^2-x;  
>plot2d(x,y,logplot=2):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.151 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-161.png

LATIHAN SOAL 1

Buatlah grafik fungsi parametrik berikut:

$$x = \frac{1 - t^2}{1 + t^2}, y = \frac{2t}{1 + t^2}, -1 \leq t \leq 1$$

```
>t=linspace(0,2*pi,100);...  
> plot2d((1-t^2)/(1+t^2),(2t)/(1+t^2)):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.152 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-163.png

Grafik tersebut akan menampilkan lingkaran dengan pusat (0,0) dan jari-jari 1.

Dapat diperhatikan bahwa kurva hanya mencakup setengah lingkaran karna rentang nilai t dibatasi dari -1 sampai 1.

LATIHAN SOAL 2

Buatlah grafik fungsi parametrik berikut:

$$x = 2.\cos(t)$$

$$y = 3.\sin(t)$$

```
>t=linspace(0,2*pi,100);...  
> plot2d(2*cos(t),3*sin(t),r=5):
```



Gambar 12.153 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-166.png

Kode diatas akan menghasilkan Grafik Fungsi Parametrik yang membentuk lingkaran dengan jari-jari 2 pada sumbu x dan jari-jari 3 pada sumbu y

Penambahan r=5 untuk melihat bahwa grafik yang terbentuk bukan lingkaran tetapi elips.

LATIHAN SOAL 3

Buatlah grafik fungsi parametrik diatas dengan bidang yang memenuhi syarat menjadi terarsir

```
>t=linspace(0,2*pi,100); x=2*cos(t); y=3*sin(t);...  
> plot2d(x,y,>filled,fillcolor=red,style="/"):
```



images/EMT4Plot2D_Shintia Nurcholis

Gambar 12.154 images/EMT4Plot2D_Shintia%20Nurcholisa_23030630071-167.png

>

RUJUKAN LENGKAP FUNGSI PLOT2D()

function plot2d (xv, yv, btest, a, b, c, d, xmin, xmax, r, n, ..
logplot, grid, frame, framecolor, square, color, thickness, style, ..
auto, add, user, delta, points, addpoints, pointstyle, bar, histogram, ..
distribution, even, steps, own, adaptive, hue, level, contour, ..
nc, filled, fillcolor, outline, title, xl, yl, maps, contourcolor, ..
contourwidth, ticks, margin, clipping, cx, cy, insimg, spectral, ..
cgrid, vertical, smaller, dl, niveau, levels)

Multipurpose plot function for plots in the plane (2D plots). This function can do plots of functions of one variables, data plots, curves in the plane, bar plots, grids of complex numbers, and implicit plots of functions of two variables.

Parameters

x,y : equations, functions or data vectors

a,b,c,d : Plot area (default a=-2,b=2)

r : if r is set, then a=cx-r, b=cx+r, c=cy-r, d=cy+r

r can be a vector [rx,ry] or a vector [rx1,rx2,ry1,ry2]

xmin,xmax : range of the parameter for curves

auto : Determine y-range automatically (default)

square : if true, try to keep square x-y-ranges

n : number of intervals (default is adaptive)

grid : 0 = no grid and labels,

1 = axis only,

2 = normal grid (see below for the number of colors)

3 = inside axis

4 = no grid

5 = full grid including margin

6 = ticks at the frame

7 = axis only

8 = axis only, sub-ticks

frame : 0 = no frame

framecolor: color of the frame and the grid

margin : number between 0 and 0.4 for the margin around the
plot

color : Color of curves. If this is a vector of colors,

it will be used for each row of a matrix of plots.

point plots, it should be a column vector. If a row

full matrix of colors is used for point plots, it

each data point.

thickness : line thickness for curves

This value can be smaller than 1 for very thin lines

style : Plot style for lines, markers, and fills.

For points use

"[]", "<>", ".", "..", "...",

"*", "+", "|", "-", "o"

"[]#", "<>#", "o#" (filled shapes)

"[]w", "<>w", "ow" (non-transparent)

For lines use

"-", "--", "-.", ".", "-.", "-.", ">"

For filled polygons or bar plots use

"#", "#o", "o", "/", "\", "\/",

"+", "|", "-", "t"

points : plot single points instead of line segments

addpoints : if true, plots line segments and points

add : add the plot to the existing plot

user : enable user interaction for functions

delta : step size for user interaction

bar : bar plot (x are the interval bounds, y the interval values)

histogram : plots the frequencies of x in n subintervals

distribution=n : plots the distribution of x with n subintervals

even : use inter values for automatic histograms.

steps : plots the function as a step function (steps=1,2)

adaptive : use adaptive plots (n is the minimal number of steps)

level : plot level lines of an implicit function of two variables

outline : draws boundary of level ranges.

If the level value is a 2xn matrix, ranges of levels will be drawn in the color using the given fill style. If outline is true, it will be drawn in the contour color. Using this feature, regions of $f(x,y)$ between limits can be marked.

hue : add hue color to the level plot to indicate the function

value

contour : Use level plot with automatic levels

nc : number of automatic level lines

title : plot title (default “”)

xl, yl : labels for the x- and y-axis

smaller : if >0, there will be more space to the left for labels.

vertical :

Turns vertical labels on or off. This changes the global variable `verticallabels` locally for one plot. The value 1 sets only vertical text, the value 2 uses vertical numerical labels on the y axis.

filled : fill the plot of a curve

fillcolor : fill color for bar and filled curves

outline : boundary for filled polygons

logplot : set logarithmic plots

1 = logplot in y,

2 = logplot in xy,

3 = logplot in x

own :

A string, which points to an own plot routine. With `>user`, you get

the same user interaction as in `plot2d`. The range will be set before each call to your function.

maps : map expressions (0 is faster), functions are always mapped.

contourcolor : color of contour lines

contourwidth : width of contour lines

clipping : toggles the clipping (default is true)

title :

This can be used to describe the plot. The title will appear above the plot. Moreover, a label for the x and y axis can be added with `xl="string"` or `yl="string"`. Other labels can be added with the functions `label()` or `labelbox()`. The title can be a unicode string or an image of a Latex formula.

cgrid :

Determines the number of grid lines for plots of complex grids. Should be a divisor of the the matrix size minus 1 (number of subintervals). `cgrid` can be a vector `[cx,cy]`.

Overview

The function can plot

- expressions, call collections or functions of one variable,
- parametric curves,
- x data against y data,
- implicit functions,
- bar plots,
- complex grids,
- polygons.

If a function or expression for xv is given, `plot2d()` will compute values in the given range using the function or expression. The expression must be an expression in the variable x . The range must

be defined in the parameters a and b unless the default range $[-2,2]$ should be used. The y -range will be computed automatically,

unless c and d are specified, or a radius r , which yields the range $[-r,r]$ for x and y . For plots of functions, `plot2d` will use an adaptive evaluation of the function by default. To speed up the plot for complicated functions, switch this off with `<adaptive`, and

optionally decrease the number of intervals n . Moreover, `plot2d()`

will by default use mapping. I.e., it will compute the plot element

for element. If your expression or your functions can handle a vector x , you can switch that off with `<maps` for faster evaluation.

Note that adaptive plots are always computed element for element.

If functions or expressions for both xv and for yv are specified, `plot2d()` will compute a curve with the xv values as x -coordinates

and the yv values as y -coordinates. In this case, a range should be

defined for the parameter using $xmin$, $xmax$. Expressions contained

in strings must always be expressions in the parameter variable x .

unicodehyperref hyphensurl []book xcolor amsmath,amssymb
iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype
[protrusion]basicmath parskip graphicx bookmark xurl

MENGGAMBAR PLOT 3D DENGAN EMT

Ini adalah pengenalan plot 3D di Euler. Kita memerlukan plot 3D untuk memvisualisasikan fungsi dua variabel.

Euler menggambar fungsi tersebut menggunakan algoritma pengurutan untuk menyembunyikan bagian di latar belakang. Secara umum Euler menggunakan proyeksi sentral. Defaultnya adalah dari kuadran x-y positif menuju titik asal $x=y=z=0$, tetapi sudut $=0^\circ$ dilihat dari arah sumbu y. Sudut pandang dan ketinggian dapat diubah.

Euler bisa merencanakan

- permukaan dengan garis penetasan dan level atau rentang level, awan titik, kurva parametrik, permukaan implisit.

Plot 3D suatu fungsi menggunakan plot3d. Cara termudah adalah dengan memplot ekspresi dalam x dan y. Parameter r mengatur rentang plot sekitar (0,0).

```
>aspect(1.5); plot3d("x^2+sin(y)",-5,5,0,6*pi):
```

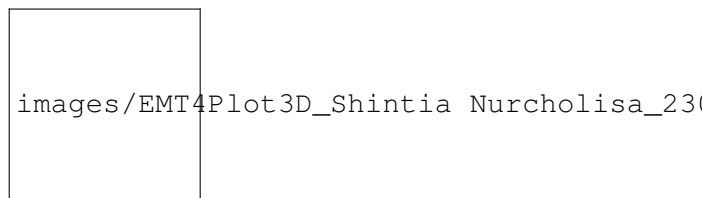


images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.155 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-001.png

```
>plot3d("x^2+x*sin(y)",-5,5,0,6*pi):
```

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.



Gambar 12.156 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-002.png

FUNGSI DUA VARIABEL

Untuk grafik suatu fungsi, gunakan

- ekspresi sederhana dalam x dan y,
- nama fungsi dari dua variabel
- atau matriks data.

Standarnya adalah kisi-kisi kawat berisi dengan warna berbeda di kedua sisi. Perhatikan bahwa jumlah interval kisi default adalah 10, tetapi plot menggunakan jumlah default persegi panjang 40x40 untuk membuat permukaannya. Ini bisa diubah.

- $n=40$, $n=[40,40]$: jumlah garis kisi di setiap arah
- $grid=10$, $grid=[10,10]$: jumlah garis grid di setiap arah.

Kami menggunakan default $n=40$ dan $grid=10$.

`>plot3d("x2+y2"):`



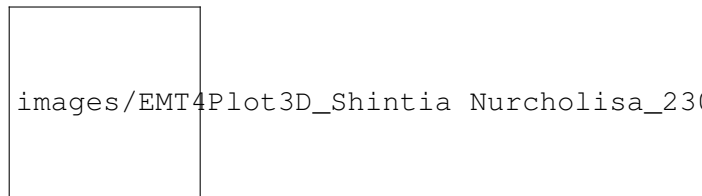
Gambar 12.157 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-003.png

Interaksi pengguna dimungkinkan dengan parameter `>pengguna`. Pengguna dapat menekan tombol berikut.

- kiri, kanan, atas, bawah: memutar sudut pandang
- +,-: memperbesar atau memperkecil

- a: menghasilkan anaglyph (lihat di bawah)
- l : tombol nyalakan sumber cahaya (lihat dibawah)
- spasi: reset ke default
- kembali: akhiri interaksi

```
>plot3d("exp(-x2+y2)",>user,...
> title="Turn with the vector keys (press return to finish)");
```



Gambar 12.158 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-004.png

Rentang plot untuk fungsi dapat ditentukan dengan

- a,b: rentang x
- c,d: rentang y
- r : persegi simetris di sekitar (0,0).
- n : jumlah subinterval untuk plot.

Ada beberapa parameter untuk menskalakan fungsi atau mengubah tampilan grafik.

fscale: menskalakan ke nilai fungsi (defaultnya adalah <fscale).

skala: angka atau vektor 1x2 untuk menskalakan ke arah x dan y.

bingkai: jenis bingkai (default 1).

```
>plot3d("exp(-(x2+y2)/5)",r=10,n=80,fscale=4,scale=1.2,frame=3,>user):
```

Tampilan dapat diubah dengan berbagai cara.

- Jarak: jarak pandang ke plot.



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.159 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-005.png

- zoom: nilai zoom.
- sudut: sudut terhadap sumbu y negatif dalam radian.
- tinggi: ketinggian pandangan 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.

Pada contoh berikut, `sudut=0` dan `tinggi=0` dilihat dari sumbu y negatif. Label sumbu untuk `y` disembunyikan dalam kasus ini.

```
>plot3d("x^2+y",distance=3,zoom=1,angle=pi/2,height=0):
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.160 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-006.png

Plot selalu terlihat berada di tengah kubus plot. Anda dapat memindahkan bagian tengah dengan parameter `tengah`.


```
>plot3d("x4+y2",a=0,b=1,c=-1,d=1,angle=-20°,height=20°, ...
> center=[0.4,0,0],zoom=5):
```



images/EMT4Plot3D_Shintia Nurcholisa_23030630071-007.png

Gambar 12.161 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-007.png

Plotnya diskalakan agar sesuai dengan unit kubus untuk dilihat. Jadi tidak perlu mengubah jarak atau zoom tergantung ukuran plot. Namun labelnya mengacu pada ukuran sebenarnya.

Jika Anda mematakannya dengan `scale=false`, Anda harus berhati-hati agar plot tetap masuk ke dalam jendela plotting, dengan mengubah jarak pandang atau zoom, dan memindahkan bagian tengah.

```
>plot3d("5*exp(-x2-y2)",r=2,<fscale,<scale,distance=13,height=50°,
...
> center=[0,0,-2],frame=3):
```

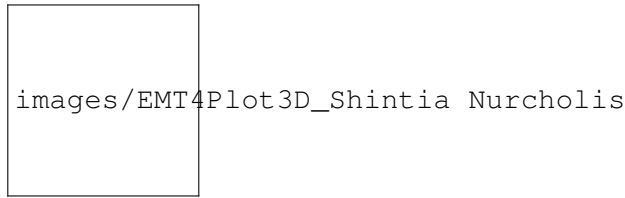


images/EMT4Plot3D_Shintia Nurcholisa_23030630071-008.png

Gambar 12.162 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-008.png

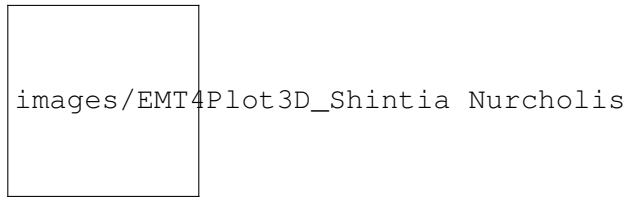
Plot kutub juga tersedia. Parameter `polar=true` menggambar plot kutub. Fungsi tersebut harus tetap merupakan fungsi dari x dan y . Parameter `"fscale"` menskalakan fungsi dengan skalanya sendiri. Kalau tidak, fungsinya akan diskalakan agar sesuai dengan kubus.

```
>plot3d("1/(x2+y2+1)",r=5,>polar, ...
> fscale=2,>hue,n=100,zoom=4,>contour,color=blue):
```



Gambar 12.163 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-009.png

```
>function f(r) := exp(-r/2)*cos(r); ...  
> plot3d("f(x2+y2)",>polar,scale=[1,1,0.4],r=pi,frame=3,zoom=4):
```

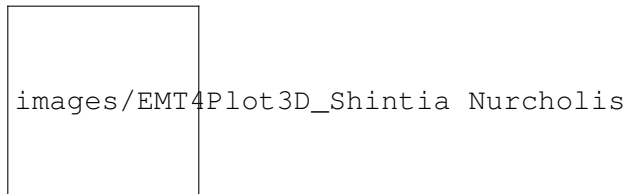


Gambar 12.164 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-010.png

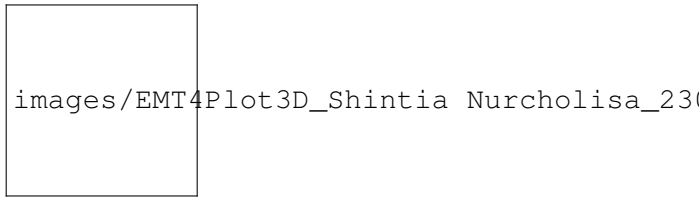
Parameter memutar memutar fungsi di x di sekitar sumbu x.

- putar=1: Menggunakan sumbu x
- putar=2: Menggunakan sumbu z

```
>plot3d("x2+1",a=-1,b=1,rotate=true,grid=5):
```

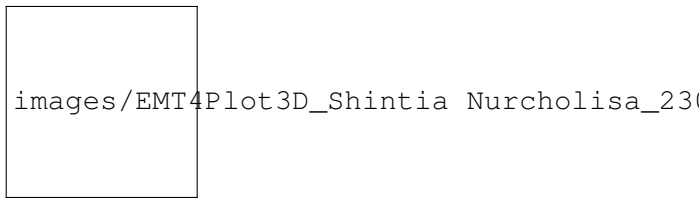


Gambar 12.165 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-011.png



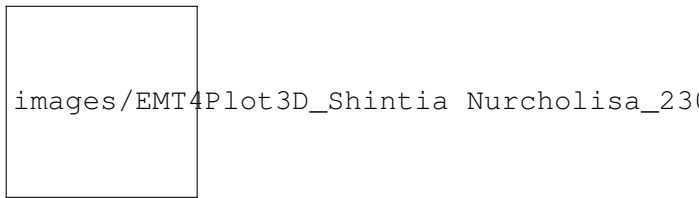
Gambar 12.166 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-012.png

```
>plot3d("x^2+1",a=-1,b=1,rotate=2,grid=5):  
>plot3d("sqrt(25-x^2)",a=0,b=5,rotate=1):
```



Gambar 12.167 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-013.png

```
>plot3d("x*sin(x)",a=0,b=6pi,rotate=2):
```



Gambar 12.168 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-014.png

Berikut adalah plot dengan tiga fungsi.

```
>plot3d("x","x^2+y^2","y",r=2,zoom=3.5,frame=3):
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.169 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-015.png

PLOT KONTUR

Untuk plotnya, Euler menambahkan garis grid. Sebaliknya dimungkinkan untuk menggunakan garis datar dan rona satu warna atau rona warna spektral. Euler dapat menggambar ketinggian fungsi pada plot dengan arsiran. Di semua plot 3D, Euler dapat menghasilkan anaglyph merah/cyan.

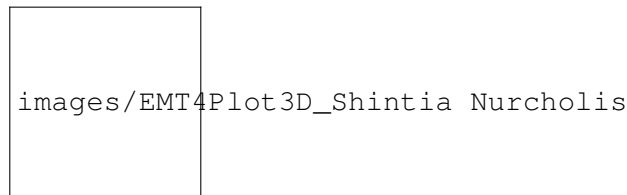
- >hue: Mengaktifkan bayangan cahaya, bukan kabel.
- >kontur: Membuat plot garis kontur otomatis pada plot.

- level=. . . (atau level): Vektor nilai 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 dengan ukuran 100x100 titik, menskalakan fungsi dan plot, dan menggunakan sudut pandang yang berbeda.

```
>plot3d("exp(-x2-y2)",r=2,n=100,level="thin",...  
> >contour,>spectral,fscale=1,scale=1.1,angle=45°,height=20°):
```

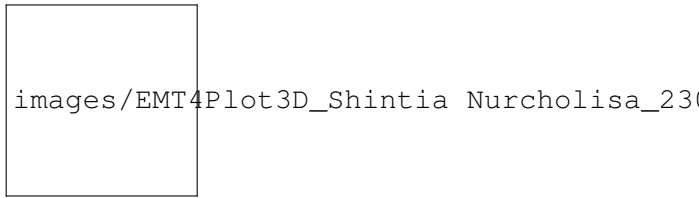


Gambar 12.170 images/EMT4Plot3D_Shintia%20Nurcholis_23030630071-016.png

```
>plot3d("exp(x*y)",angle=100°,>contour,color=green):
```

Bayangan defaultnya menggunakan warna abu-abu. Namun rentang warna spektral juga tersedia.

- >spektral: Menggunakan skema spektral default

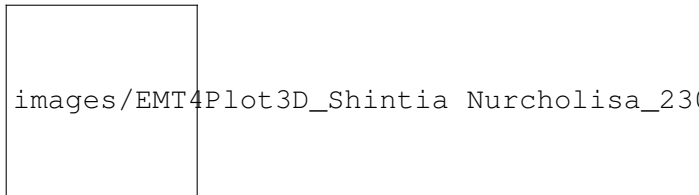


Gambar 12.171 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-017.png

- `color=...`: Menggunakan warna khusus atau skema spektral

Untuk plot berikut, kami menggunakan skema spektral default dan menambah jumlah titik untuk mendapatkan tampilan yang sangat mulus.

```
>plot3d("x2+y2",>spectral,>contour,n=100):
```



Gambar 12.172 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-018.png

Selain garis level otomatis, kita juga dapat menetapkan nilai garis level. Ini akan menghasilkan garis level yang tipis, bukan rentang level.

```
>plot3d("x2+y2",0,5,0,5,level=-1:0.1:1,color=redgreen):
```

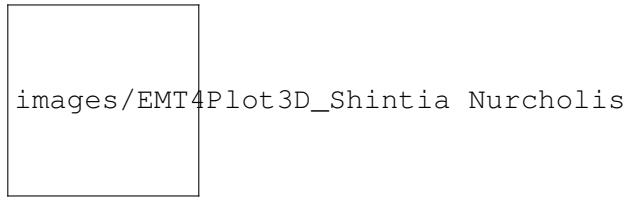
Dalam plot berikut, kita menggunakan dua pita tingkat yang sangat luas dari -0,1 hingga 1, dan dari 0,9 hingga 1. Ini dimasukkan sebagai matriks dengan batas tingkat sebagai kolom.

Selain itu, kami melapisi grid dengan 10 interval di setiap arah.

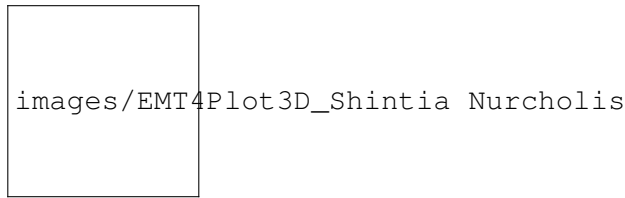
```
>plot3d("x2+y3",level=[-0.1,0.9;0,1],...
```

```
>>spectral,angle=30°,grid=10,contourcolor=gray):
```

Pada contoh berikut, kita memplot himpunan, di mana



Gambar 12.173 images/EMT4Plot3D_Shintia%20Nurcholis_23030630071-019.png

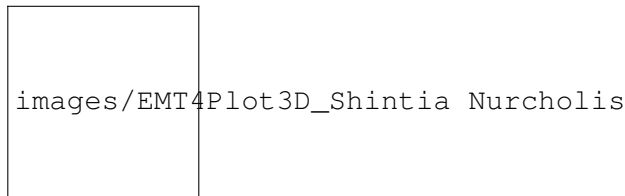


Gambar 12.174 images/EMT4Plot3D_Shintia%20Nurcholis_23030630071-020.png

$$f(x, y) = x^y - y^x = 0$$

Kami menggunakan satu garis tipis untuk garis level.

```
>plot3d("x^y-y^x",level=0,a=0,b=6,c=0,d=6,contourcolor=red,n=100):
```

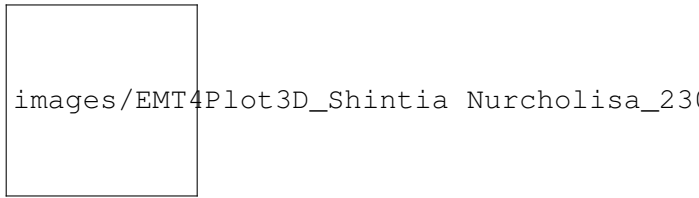


Gambar 12.175 images/EMT4Plot3D_Shintia%20Nurcholis_23030630071-022.png

Dimungkinkan untuk menampilkan bidang kontur di bawah plot. Warna dan jarak ke plot dapat ditentukan.

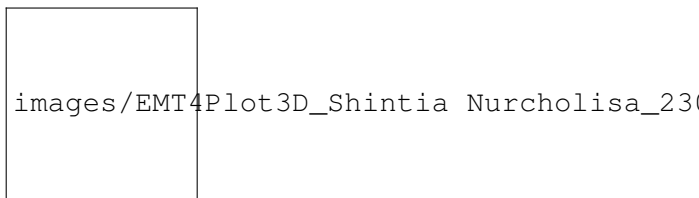
```
>plot3d("x^2+y^4",>cp,cpcolor=green,cpdelta=0.2):
```

Berikut beberapa gaya lainnya. Kami selalu mematikan bingkai, dan menggunakan berbagai skema warna untuk plot dan kisi.



Gambar 12.176 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-023.png

```
>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):
```



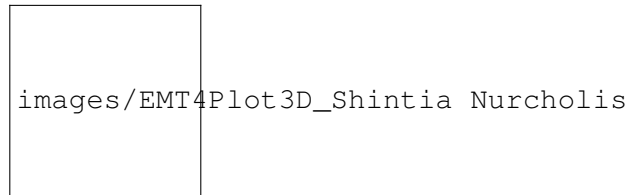
Gambar 12.177 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-024.png

Ada beberapa skema spektral lainnya, yang diberi nomor dari 1 hingga 9. Namun Anda juga dapat menggunakan warna=nilai, di mana nilai

- spektral: untuk rentang dari biru hingga merah

- putih: untuk rentang yang lebih redup
- kuningbiru, unguhijau, birukuning, hijaumerah
- birukuning, hijauungu, kuningbiru, merahhijau

```
>figure(3,3); ...
> for i=1:9; ...
> figure(i); plot3d("x2+y2",spectral=i,>contour,>cp,<frame,zoom=4);
...
> end; ...
> figure(0):
```



Gambar 12.178 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-025.png

Sumber cahaya dapat diubah dengan `l` dan tombol kursor selama interaksi pengguna. Itu juga dapat diatur dengan parameter.

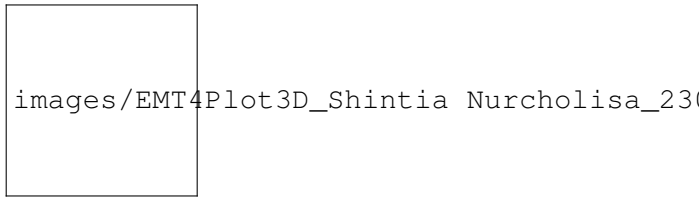
- cahaya : arah datangnya cahaya
- amb: cahaya sekitar antara 0 dan 1

Perhatikan bahwa program ini tidak membuat perbedaan antara sisi plot. Tidak ada bayangan. Untuk ini, Anda memerlukan Povray.

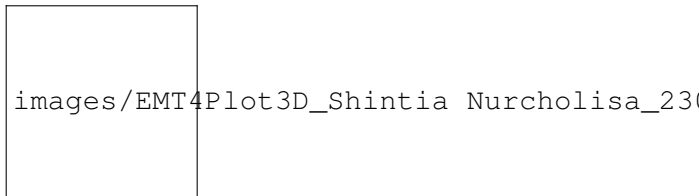
```
>plot3d("-x2-y2", ...
> hue=true,light=[0,1,1],amb=0,user=true, ...
> title="Press l and cursor keys (return to exit)");
```

Parameter warna mengubah warna permukaan. Warna garis level juga bisa 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):
```



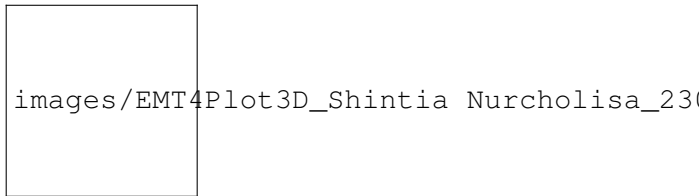
Gambar 12.179 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-026.png



Gambar 12.180 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-027.png

Warna 0 memberikan efek pelangi yang istimewa.

```
>plot3d("x2/(x2+y2+1)",color=0,hue=true,grid=10):
```



Gambar 12.181 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-028.png

Permukaannya juga bisa transparan.

```
>plot3d("x2+y2",>transparent,grid=10,wirecolor=red):
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.182 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-029.png

PLOT IMPLISIT

Ada juga plot implisit dalam tiga dimensi. Euler menghasilkan pemotongan melalui objek. Fitur plot3d mencakup plot implisit. Plot ini menunjukkan himpunan nol suatu fungsi dalam tiga variabel.

Solusi dari

$$f(x, y, z) = 0$$

dapat divisualisasikan dalam potongan yang sejajar dengan bidang x-y-, x-z- dan y-z.

- implisit=1: dipotong sejajar bidang y-z
- implisit=2: dipotong sejajar dengan bidang x-z
- implisit=4: dipotong sejajar bidang x-y

Tambahkan nilai-nilai ini, jika Anda mau. Dalam contoh kita memplot

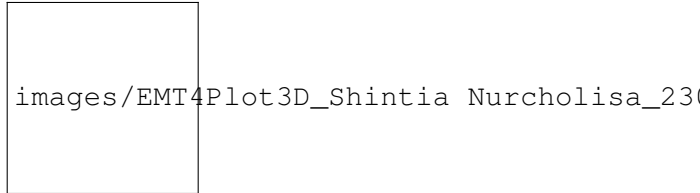
$$M = \{(x, y, z) : x^2 + y^3 + zy = 1\}$$

>plot3d("x^{2+y}3+z*y-1",r=5,implicit=3):

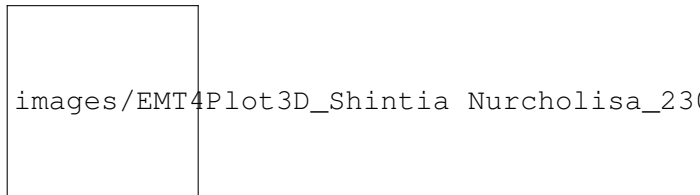


Gambar 12.183 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-032.png

```
>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):
>plot3d("x2+y2+4*x*z+z3",>implicit,r=2,zoom=2.5):
```



Gambar 12.184 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-033.png



Gambar 12.185 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-034.png

MERENCANAKAN DATA 3D

Sama seperti plot2d, plot3d menerima data. Untuk objek 3D, Anda perlu menyediakan matriks nilai x-, y- dan z, atau tiga fungsi atau ekspresi $f_x(x,y)$, $f_y(x,y)$, $f_z(x,y)$.

$$\gamma(t, s) = (x(t, s), y(t, s), z(t, s))$$

Karena x,y,z adalah matriks, kita asumsikan bahwa (t,s) melewati grid 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, kita menggunakan vektor nilai t dan vektor kolom nilai s untuk membuat parameter permukaan bola. Dalam gambar kita dapat menandai wilayah, dalam kasus kita wilayah kutub.

```
>t=linspace(0,2pi,180); s=linspace(-pi/2,pi/2,90)'; ...
> x=cos(s)*cos(t); y=cos(s)*sin(t); z=sin(s); ...
> plot3d(x,y,z,>hue, ...
> color=blue,<frame,grid=[10,20], ...
> values=s,contourcolor=red,level=[90°-24°;90°-22°], ...
> scale=1.4,height=50°):
```

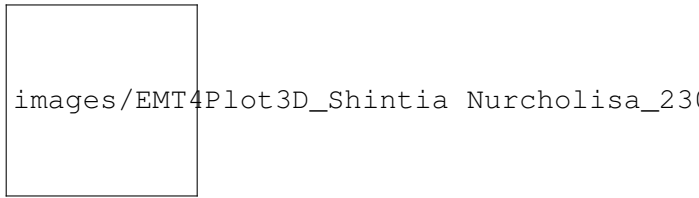


images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.186 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-036.png

Berikut ini contohnya yaitu grafik suatu fungsi.

```
>t=-1:0.1:1; s=(-1:0.1:1)'; plot3d(t,s,t*s,grid=10):
```

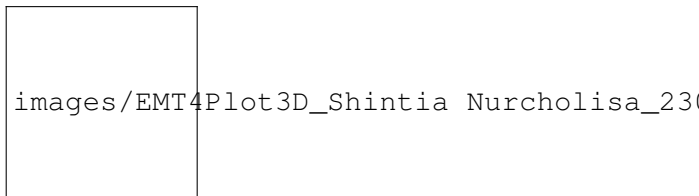



Gambar 12.187 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-037.png

Namun, kita bisa membuat berbagai macam permukaan. Berikut adalah permukaan yang sama sebagai suatu fungsi

$$x = yz$$

>plot3d(t*s,t,s,angle=180°,grid=10):



Gambar 12.188 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-039.png

Dengan lebih banyak usaha, kita dapat menghasilkan banyak permukaan.

Dalam contoh berikut kita membuat tampilan bayangan dari bola yang terdistorsi. Koordinat bola yang biasa adalah

$$\gamma(t, s) = (\cos(t) \cos(s), \sin(t) \sin(s), \cos(s))$$

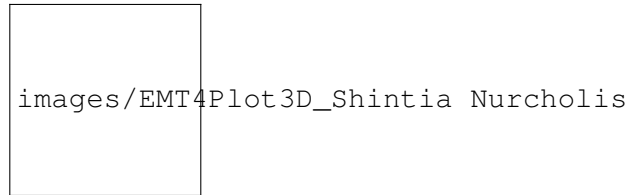
dengan

$$0 \leq t \leq 2\pi, \quad -\frac{\pi}{2} \leq s \leq \frac{\pi}{2}.$$

Kami mendistorsi ini dengan sebuah faktor

$$d(t, s) = \frac{\cos(4t) + \cos(8s)}{4}.$$

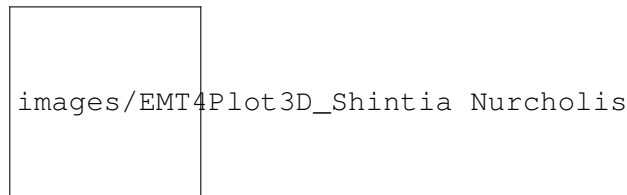
```
>t=linspace(0,2pi,320); s=linspace(-pi/2,pi/2,160)'; ...
> d=1+0.2*(cos(4*t)+cos(8*s)); ...
> plot3d(cos(t)*cos(s)*d,sin(t)*cos(s)*d,sin(s)*d,hue=1, ...
> light=[1,0,1],frame=0,zoom=5):
```



Gambar 12.189 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-043.png

Tentu saja, point cloud juga dimungkinkan. Untuk memplot data titik dalam ruang, kita memerlukan tiga vektor untuk koordinat titik-titik tersebut.

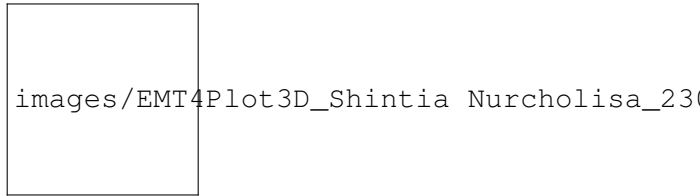
```
Gayanya sama seperti di plot2d dengan points=true;
>n=500; ...
> plot3d(normal(1,n),normal(1,n),normal(1,n),points=true,style="."):
```



Gambar 12.190 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-044.png

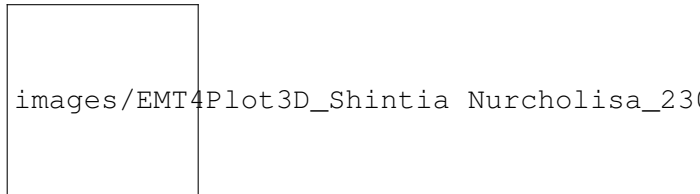
Dimungkinkan juga untuk memplot kurva dalam 3D. Dalam hal ini, lebih mudah untuk menghitung terlebih dahulu titik-titik kurva. Untuk kurva pada bidang kita menggunakan barisan koordinat dan parameter wire=true.

```
>t=linspace(0,8pi,500); ...
> plot3d(sin(t),cos(t),t/10,>wire,zoom=3):
```



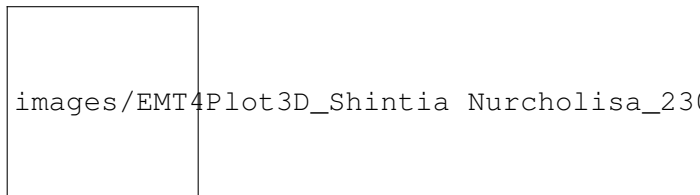
Gambar 12.191 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-045.png

```
>t=linspace(0,4pi,1000); plot3d(cos(t),sin(t),t/2pi,>wire, ...
> linewidth=3,wirecolor=blue):
```



Gambar 12.192 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-046.png

```
>X=cumsum(normal(3,100)); ...
> plot3d(X[1],X[2],X[3],>anaglyph,>wire):
```



Gambar 12.193 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-047.png

EMT juga dapat membuat plot dalam mode anaglyph. Untuk melihat plot seperti itu, Anda memerlukan kacamata berwarna merah/sian.

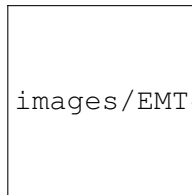
```
>plot3d("x2+y3",>anaglyph,>contour,angle=30°):
```



Gambar 12.194 images/EMT4Plot3D_Shintia%20Nurcholis_23030630071-048.png

Seringkali skema warna spektral digunakan untuk plot. Ini menekankan ketinggian fungsinya.

```
>plot3d("x2*y3-y",>spectral,>contour,zoom=3.2):
```



Gambar 12.195 images/EMT4Plot3D_Shintia%20Nurcholis_23030630071-049.png

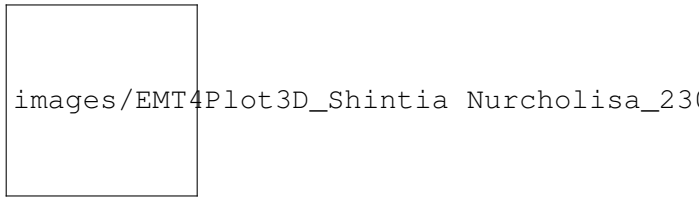
Euler juga dapat memplot permukaan yang diparameterisasi, jika parameternya adalah nilai x , y , dan z dari gambar kotak persegi panjang di ruang tersebut.

Untuk demo berikut, kami menyiapkan parameter u - dan v -, dan menghasilkan koordinat ruang dari parameter tersebut.

```
>u=linspace(-1,1,10); v=linspace(0,2*pi,50)'; ...
> X=(3+u*cos(v/2))*cos(v); Y=(3+u*cos(v/2))*sin(v); Z=u*sin(v/2);
...
> plot3d(X,Y,Z,>anaglyph,<frame,>wire,scale=2.3):
```

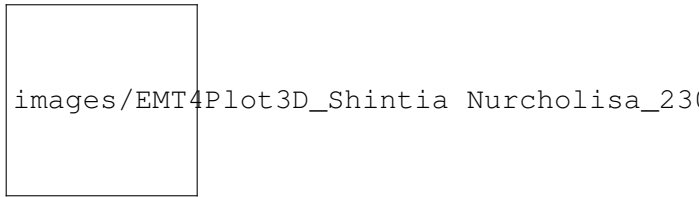
Berikut adalah contoh yang lebih rumit, yang megah dengan kaca mata merah/sian.

```
>u:=linspace(-pi,pi,160); v:=linspace(-pi,pi,400)'; ...
> x:=(4*(1+.25*sin(3*v))+cos(u))*cos(2*v); ...
```



Gambar 12.196 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-050.png

```
> y:=(4*(1+.25*sin(3*v))+cos(u))*sin(2*v); ...  
> z=sin(u)+2*cos(3*v); ...  
> plot3d(x,y,z,frame=0,scale=1.5,hue=1,light=[1,0,-1],zoom=2.8,>anaglyph):
```



Gambar 12.197 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-051.png

PLOT STATISTIK

Plot batang juga dimungkinkan. Untuk itu, kita harus menyediakannya

- x: vektor baris dengan $n+1$ elemen
- y: vektor kolom dengan $n+1$ elemen
- z: matriks nilai $n \times n$.

z bisa lebih besar, tetapi hanya nilai $n \times n$ yang akan digunakan.

Dalam contoh ini, pertama-tama kita menghitung nilainya. Kemudian kita sesuaikan x dan y, sehingga vektor-vektornya berpusat pada nilai yang digunakan.

```
>x=-1:0.1:1; y=x'; z=x2+y2; ...  
> xa=(x|1.1)-0.05; ya=(y|1.1)-0.05; ...  
> plot3d(xa,ya,z,bar=true):
```

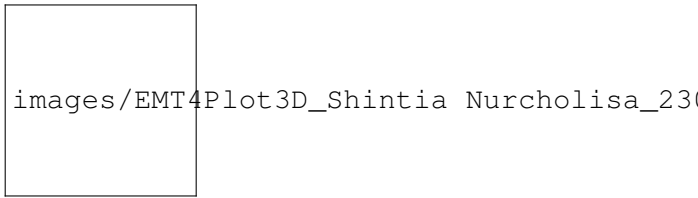


Gambar 12.198 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-052.png

Dimungkinkan untuk membagi plot suatu permukaan menjadi dua bagian atau lebih.

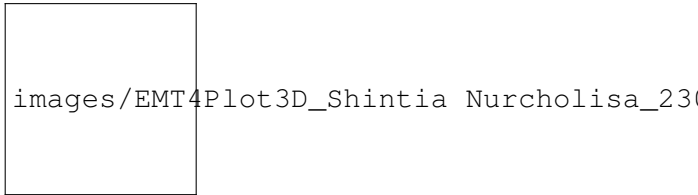
```
>x=-1:0.1:1; y=x'; z=x+y; d=zeros(size(x)); ...  
> plot3d(x,y,z,disconnect=2:2:20):
```

Jika memuat atau menghasilkan matriks data M dari file dan perlu memplotnya dalam 3D, Anda dapat menskalakan matriks ke $[-1,1]$ dengan `skala(M)`, atau menskalakan matriks dengan `>zscale`. Hal ini dapat dikombinasikan dengan faktor penskalaan individual yang diterapkan sebagai tambahan.



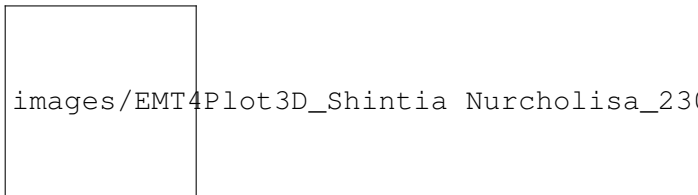
Gambar 12.199 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-053.png

```
>i=1:20; j=i'; ...
> plot3d(i*j^2+100*normal(20,20),>zscale,scale=[1,1,1.5],angle=-40°,zoom=1.8):
```



Gambar 12.200 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-054.png

```
>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]):
```



Gambar 12.201 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-055.png

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/EMT4Plot3D_Shintia Nurcholis

Gambar 12.202 images/EMT4Plot3D_Shintia%20Nurcholis_23030630071-056.png

```
>ekspresi &= (x^2+y^2-1)^3-x^2*y^3; $ekspresi
```

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

Kami ingin memutar kurva hati di sekitar sumbu y. Inilah ungkapan yang mendefinisikan hati:

$$f(x, y) = (x^2 + y^2 - 1)^3 - x^2 y^3.$$

Selanjutnya kita atur

$$x = r \cos(a), \quad y = r \sin(a).$$

```
>function fr(r,a) &= ekspresi with [x=r*cos(a),y=r*sin(a)] | trigreduce;
$fr(r,a)
```

$$(r^2 - 1)^3 + \frac{(\sin(5a) - \sin(3a) - 2 \sin a) r^5}{16}$$

Hal ini memungkinkan untuk mendefinisikan fungsi numerik, yang menyelesaikan r, jika a diberikan. Dengan fungsi tersebut kita dapat memplot jantung yang diputar sebagai permukaan parametrik.


```

>function map f(a) := bisect("fr",0,2;a); ...
> t=linspace(-pi/2,pi/2,100); r=f(t); ...
> s=linspace(pi,2pi,100)'; ...
> plot3d(r*cos(t)*sin(s),r*cos(t)*cos(s),r*sin(t), ...
> >hue,<frame,color=red,zoom=4,amb=0,max=0.7,grid=12,height=50°):

```



images/EMT4Plot3D_Shintia Nurcholisa_23030630071-

Gambar 12.203 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-061.png

Berikut ini adalah plot 3D dari gambar di atas yang diputar mengelilingi sumbu z. Kami mendefinisikan fungsi yang mendeskripsikan objek.

```

>function f(x,y,z) ...

r=x^2+y^2;
return (r+z^2-1)^3-r*z^3;
endfunction

>plot3d("f(x,y,z)", ...
> xmin=0,xmax=1.2,ymin=-1.2,ymax=1.2,zmin=-1.2,zmax=1.4, ...
> implicit=1,angle=-30°,zoom=2.5,n=[10,100,60],>anaglyph):

```



images/EMT4Plot3D_Shintia Nurcholisa_23030630071-

Gambar 12.204 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-062.png

PLOT 3D KHUSUS

Fungsi `plot3d` bagus untuk dimiliki, tetapi tidak memenuhi semua kebutuhan. Selain rutinitas yang lebih mendasar, dimungkinkan untuk mendapatkan plot berbingkai dari objek apa pun yang Anda sukai.

Meskipun Euler bukan program 3D, ia dapat menggabungkan beberapa objek dasar. Kami mencoba memvisualisasikan paraboloid dan garis singgungnya.

```
>function myplot ...  
  
y=-1:0.01:1; x=(-1:0.01:1)';  
plot3d(x,y,0.2*(x-0.1)/2,<scale,<frame,>hue, ..  
    hues=0.5,>contour,color=orange);  
h=holding(1);  
plot3d(x,y,(x^2+y^2)/2,<scale,<frame,>contour,>hue);  
holding(h);  
endfunction
```

Sekarang `framedplot()` menyediakan bingkai, dan mengatur tampilan.

```
>framedplot("myplot",[-1,1,-1,1,0,1],height=0,angle=-30°,...  
> center=[0,0,-0.7],zoom=3):
```



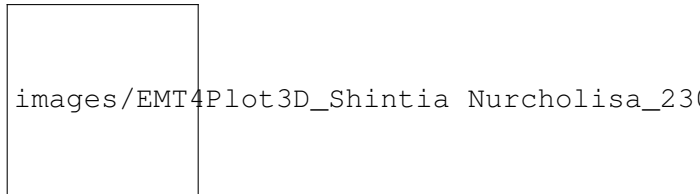
images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.205 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-063.png

Dengan cara yang sama, Anda dapat memplot bidang kontur secara manual. Perhatikan bahwa `plot3d()` menyetel jendela ke `fullwindow()` secara default, tetapi `plotcontourplane()` berasumsi demikian.

```
>x=-1:0.02:1.1; y=x'; z=x2-y4;  
>function myplot (x,y,z) ...  
>
```

```
>myplot(x,y,z):
```



Gambar 12.206 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-064.png

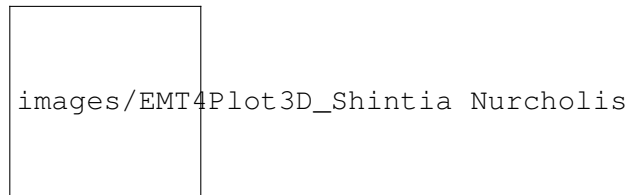
ANIMASI

Euler dapat menggunakan frame untuk melakukan pra-komputasi animasi.

Salah satu fungsi yang memanfaatkan teknik ini adalah memutar. Itu dapat mengubah sudut pandang dan menggambar ulang plot 3D. Fungsi ini memanggil `addpage()` untuk setiap plot baru. Akhirnya ia menganimasikan plotnya.

Silakan pelajari sumber rotasi untuk melihat lebih detail.

```
>function testplot () := plot3d("x2+y3"); ...  
> rotate("testplot"); testplot():
```



Gambar 12.207 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-065.png

MENGGAMBAR POVRAY

Dengan bantuan file Euler povray.e, Euler dapat membuat 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 dalam jalur lingkungan, atau mengatur variabel "defaultpovray" dengan jalur lengkap yang mengarah ke "pvengine.exe".><http://www.povray.org/> dan meletakkan subdirektori "bin" dari Povray ke dalam jalur lingkungan, atau mengatur variabel "defaultpovray" dengan jalur lengkap yang mengarah ke "pvengine.exe".

Antarmuka Povray dari Euler membuat 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 membuat 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. Fungsi 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 buku catatan Euler.

Selain pov3d(), ada banyak fungsi, yang menghasilkan objek Povray. Fungsi-fungsi ini mengembalikan string, yang berisi kode Povray untuk objek. Untuk menggunakan fungsi-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 mulai, dan PNG akan dimasukkan ke dalam buku catatan Euler.

Fungsi objek memiliki parameter yang disebut "look", yang memerlukan string dengan kode Povray untuk tekstur dan penyelesaian objek. Fungsi povlook() dapat digunakan untuk menghasilkan string ini. Fungsi 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 sumbu x, y, z dalam arah kanan.

Anda perlu memuat berkas povray.

```
>load povray;
```

Pastikan direktori bin Povray ada di jalur tersebut. Jika tidak, edit variabel berikut sehingga berisi jalur ke povray yang dapat dieksekusi.

```
>defaultpovray="C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe"
```

```
C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe
```

Untuk kesan pertama, kami membuat fungsi sederhana. Perintah berikut menghasilkan file povray di direktori pengguna Anda, dan menjalankan Povray untuk melakukan ray tracing pada file ini.

Jika Anda menjalankan 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 berjalan. Anda dapat menekan batal untuk menghentikan pertanyaan lebih lanjut. Anda mungkin harus menekan OK di jendela Povray untuk mengakui dialog awal Povray.

```
>plot3d("x2+y2",zoom=2):
```



images/EMT4Plot3D_Shintia Nurcholisa_23030630071-066.png

Gambar 12.208 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-066.png

```
>pov3d("x2+y2",zoom=3);
```

Kita dapat membuat fungsi tersebut transparan dan menambahkan penyelesaian lainnya. Kita juga dapat menambahkan garis level pada



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.209 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-067.png

plot fungsi.

```
> pov3d("x2+y3", axiscolor=red, angle=-45°, > anaglyph, ...
> look=povlook(cyan, 0.2), level=-1:0.5:1, zoom=3.8);
```



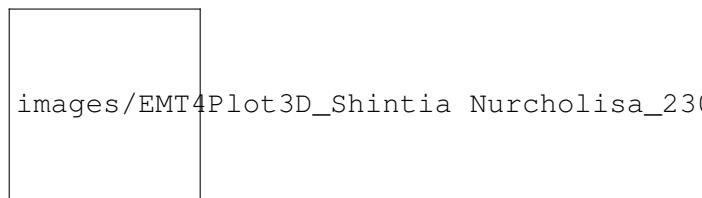
images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.210 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-068.png

Terkadang perlu untuk mencegah penskalaan fungsi, dan menskalakan fungsi secara manual.

Kami memplot himpunan titik pada bidang kompleks, di mana hasil kali jarak ke 1 dan -1 sama dengan 1.

```
> pov3d("((x-1)2+y2)*((x+1)2+y2)/40", r=2, ...
> angle=-120°, level=1/40, dlevel=0.005, light=[-1, 1, 1], height=10°, n=50,
...
> <fscale, zoom=3.8);
```

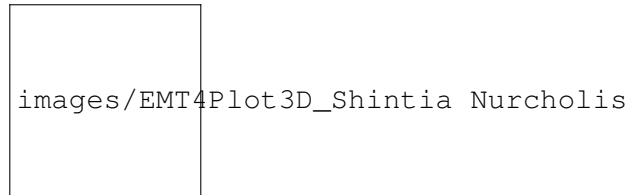
Gambar 12.211 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-069.png

MEMBUAT PLOT DENGAN KOORDINAT

Alih-alih menggunakan fungsi, kita dapat membuat plot dengan koordinat. Seperti pada `plot3d`, kita memerlukan tiga matriks untuk menentukan objek.

Dalam contoh ini, 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°,look=povlook(red,0.1),height=50°,axis=0,zoom=4,li
```



Gambar 12.212 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-070.png

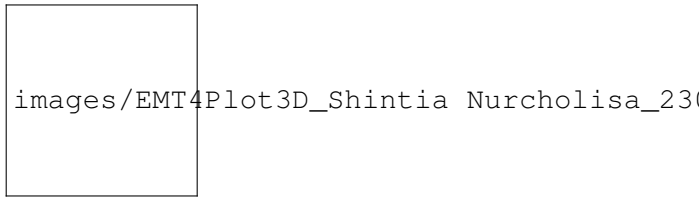
Dalam contoh berikut, kami memplot gelombang yang diredam. Kami menghasilkan gelombang dengan bahasa matriks Euler.

Kami juga menunjukkan, bagaimana objek tambahan dapat ditambahkan ke adegan `pov3d`. Untuk pembuatan objek, lihat contoh berikut. Perhatikan bahwa `plot3d` menskalakan plot, sehingga sesuai dengan kubus satuan.

```
> r=linspace(0,1,80); phi=linspace(0,2pi,80)'; ...  
> x=r*cos(phi); y=r*sin(phi); z=exp(-5*r)*cos(8*pi*r)/3; ...  
> pov3d(x,y,z,zoom=6,axis=0,height=30°,add=povsphere([0.5,0,0.25],0.15,pov  
...  
> w=500,h=300);
```

Dengan metode shading Povray yang canggih, hanya sedikit titik yang dapat menghasilkan permukaan yang sangat halus. Hanya pada batas dan bayangan, triknya mungkin menjadi jelas.

Untuk ini, kita perlu menambahkan vektor normal di setiap titik



Gambar 12.213 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-071.png

matriks.

$$Z = x^2 y^3$$

$$\begin{pmatrix} 2 & 3 \\ x & y \end{pmatrix}$$

Persamaan permukaannya adalah $[x, y, Z]$. Kita hitung dua turunan x dan y dari persamaan ini dan ambil perkalian silang sebagai normalnya.

$$dx = \text{diff}([x, y, Z], x); dy = \text{diff}([x, y, Z], y);$$

Kami mendefinisikan normal sebagai perkalian silang turunan-turunan ini dan mendefinisikan fungsi koordinat.

$$N = \text{crossproduct}(dx, dy); NX = N[1]; NY = N[2]; NZ = N[3];$$

$$\begin{pmatrix} -2xy^3, -3x^2y, 1 \end{pmatrix}$$

Kami hanya menggunakan 25 poin.

$$x = -1:0.5:1; y = x';$$

$$\text{pov3d}(x, y, Z(x, y), \text{angle}=10^\circ, \dots)$$

$$xv = NX(x, y), yv = NY(x, y), zv = NZ(x, y), <\text{shadow});$$

Berikut ini adalah simpul Trefoil yang dibuat oleh A. Busser di Povray. Ada versi yang lebih baik dari simpul ini dalam contoh-contohnya.

Trefoil Knot

Untuk tampilan yang bagus dengan tidak terlalu banyak titik, kami menambahkan vektor normal di sini. Kami menggunakan

images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.214 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-072.png

Maxima untuk menghitung nilai normal bagi kami. Pertama, tiga fungsi untuk koordinat sebagai ekspresi simbolik.

```
> X &= ((4+sin(3*y))+cos(x))*cos(2*y); ...
> Y &= ((4+sin(3*y))+cos(x))*sin(2*y); ...
> Z &= sin(x)+2*cos(3*y);
```

Kemudian dua vektor turunan ke x dan y.

```
> dx &= diff([X,Y,Z],x); dy &= diff([X,Y,Z],y);
```

Sekarang normal, yang merupakan perkalian silang dari dua turunan.

```
> dn &= crossproduct(dx,dy);
```

Sekarang mari kita evaluasi semua ini secara numerik.

```
> x:=linspace(-%pi,%pi,40); y:=linspace(-%pi,%pi,100)';
```

Vektor normal adalah evaluasi ekspresi simbolik $dn[i]$ untuk $i=1,2,3$. Sintaks untuk ini adalah `&"ekspresi"(parameter)`. Ini adalah alternatif untuk metode pada contoh sebelumnya, di mana kita mendefinisikan ekspresi simbolik NX , NY , NZ terlebih dahulu.

```
> pov3d(X(x,y),Y(x,y),Z(x,y),>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));

Kita juga dapat membuat grid dalam 3D.
> povstart(zoom=4); ...
> x=-1:0.5:1; r=1-(x+1)^2/6; ...
> t=(0°:30°:360°)'; y=r*cos(t); z=r*sin(t); ...
> writeln(povgrid(x,y,z,d=0.02,dballs=0.05)); ...
> povend();
```



Gambar 12.215 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-073.png



Gambar 12.216 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-074.png

Dengan `povgrid()`, kurva dimungkinkan.

```
> povstart(center=[0,0,1],zoom=3.6); ...  
> t=linspace(0,2,1000); r=exp(-t); ...  
> x=cos(2*pi*10*t)*r; y=sin(2*pi*10*t)*r; z=t; ...  
> writeln(povgrid(x,y,z,povlook(red))); ...  
> writeAxis(0,2,axis=3); ...  
> povend();
```



Gambar 12.217 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-075.png

OBJEK POVRAY

Di atas, kami menggunakan pov3d untuk memplot permukaan. Antarmuka povray di Euler juga dapat menghasilkan objek Povray. Objek-objek ini disimpan sebagai string di Euler, dan perlu ditulis ke berkas Povray.

Kami memulai output dengan povstart().

```
>povstart(zoom=4);
```

Pertama, kita mendefinisikan tiga silinder, dan menyimpannya dalam string di Euler.

Fungsi povx() dll. hanya mengembalikan vektor [1,0,0], yang dapat digunakan sebagai gantinya.

```
>c1=povcylinder(-povx,povx,1,povlook(red)); ...  
> c2=povcylinder(-povy,povy,1,povlook(yellow)); ...  
> c3=povcylinder(-povz,povz,1,povlook(blue)); ...  
>
```

Rangkaian tersebut berisi kode Povray, yang tidak perlu kita pahami saat itu.

```
>c2
```

```
cylinder { <0,0,-1>; <0,0,1>; 1  
  texture { pigment { color rgb <0.941176,0.941176,0.941176>  
    finish { ambient 0.2 }  
  }  
}
```

Seperti yang Anda lihat, kami menambahkan tekstur ke objek dalam tiga warna berbeda.

Hal itu dilakukan oleh povlook(), yang mengembalikan string dengan kode Povray yang relevan. Kita dapat menggunakan warna Euler default, atau menentukan warna kita sendiri. Kita juga dapat menambahkan transparansi, atau mengubah cahaya sekitar.

```
>povlook(rgb(0.1,0.2,0.3),0.1,0.5)
```

```
texture { pigment { color rgbf <0.101961,0.2,0.301961>  
  finish { ambient 0.5 }  
}
```

Sekarang kita mendefinisikan objek persimpangan dan menulis hasilnya ke berkas.

```
>writeln(povintersection([c1,c2,c3]));
```

Persimpangan tiga silinder sulit dibayangkan, jika Anda belum pernah melihatnya sebelumnya.

```
>povend;
```



images/EMT4Plot3D_Shintia Nurcholisa_23

Gambar 12.218 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-076.png

The following functions generate a fractal recursively.

The first function shows, how Euler handles simple Povray objects. The function `povbox()` returns a string, containing the box coordinates, the texture and the finish.

```
>function onebox(x,y,z,d) := povbox([x,y,z],[x+d,y+d,z+d],povlook());
```

```
>function fractal (x,y,z,h,n) ...
```

```
>
```

```
>povstart(fade=10,<shadow);
```

```
>fractal(-1,-1,-1,2,4);
```

```
>povend();
```



images/EMT4Plot3D_Shintia Nurcholisa_23

Gambar 12.219 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-077.png

Differences allow cutting off one object from another. Like intersections, there are part of the CSG objects of Povray.

```
>povstart(light=[5,-5,5],fade=10);
```

For this demonstration, we define an object in Povray, instead of using a string in Euler. Definitions are written to the file immediately.

A box coordinate of -1 just means [-1,-1,-1].

```
>povdefine("mycube",povbox(-1,1));
```

We can use this object in povobject(), which returns a string as usual.

```
>c1=povobject("mycube",povlook(red));
```

We generate a second cube, and rotate and scale it a bit.

```
>c2=povobject("mycube",povlook(yellow),translate=[1,1,1],
```

```
...
```

```
> rotate=xrotate(10°)+yrotate(10°), scale=1.2);
```

Then we take the difference of the two objects.

```
>writeln(povdifference(c1,c2));
```

Now add three axes.

```
>writeAxis(-1.2,1.2,axis=1); ...
```

```
> writeAxis(-1.2,1.2,axis=2); ...
```

```
> writeAxis(-1.2,1.2,axis=4); ...
```

```
> povend();
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.220 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-078.png

IMPLICIT FUNCTIONS

Povray can plot the set where $f(x,y,z)=0$, just like the implicit parameter in plot3d. The results looks much better, however.

The syntax for the functions is a bit different. You cannot use the output of Maxima or Euler expressions.

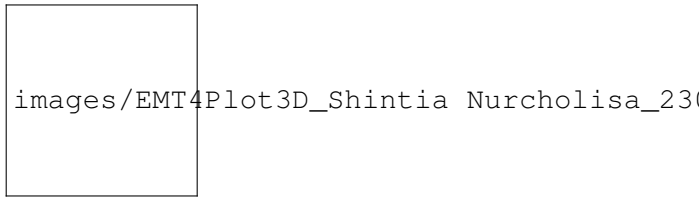
```
((x^2+y^2-c^2)^2+(z^2-1)^2)*((y^2+z^2-c^2)^2+(x^2-1)^2)*((z^2+x^2-c^2)^2+(y^2-1)^2)
>povstart(angle=70°,height=50°,zoom=4);
    >c=0.1; d=0.1; ...
> writeln(povsurface(("(pow(pow(x,2)+pow(y,2)-pow(c,2),2)+pow(pow(z,2)-1,2))*
(pow(pow(y,2)+pow(z,2)-pow(c,2),2)+pow(pow(x,2)-1,2))*
(pow(pow(z,2)+pow(c,2),2)+pow(pow(y,2)-1,2))-d",povlook(red)))); ...
> povend();
```

Error : Povray error!

Error generated by error() command

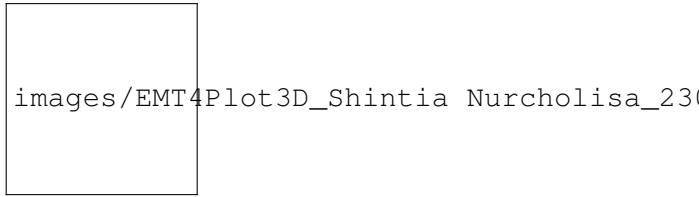
```
povray:
    error("Povray error!");
Try "trace errors" to inspect local variables after error()
povend:
    povray(file,w,h,aspect,exit);

    >povstart(angle=25°,height=10°);
    >writeln(povsurface(("pow(x,2)+pow(y,2)*pow(z,2)-1",povlook(blue),povlook(blue),
2,2,"(")));
    >povend();
    >povstart(angle=70°,height=50°,zoom=4);
    Create the implicit surface. Note the different syntax in the expression.
    >writeln(povsurface(("pow(x,2)*y-pow(y,3)-pow(z,2)",povlook(green))));
    ...
```



Gambar 12.221 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-080.png

```
> writeAxes(); ...  
> povend();
```



Gambar 12.222 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-081.png

MESH OBJECT

In this example, we show how to create a mesh object, and draw it with additional information.

We like to maximize xy under the condition $x+y=1$ and demonstrate the tangential touching of the level lines.

```
>povstart(angle=-10°,center=[0.5,0.5,0.5],zoom=7);
```

We cannot store the object in a string as before, since it is too large. So we define the object in a Povray file using `#declare`. The function `povtriangle()` does this automatically. It can accept normal vectors just like `pov3d()`.

The following defines the mesh object, and writes it immediately into the file.

```
>x=0:0.02:1; y=x'; z=x*y; vx=-y; vy=-x; vz=1;
```

```
>mesh=povtriangles(x,y,z,““,vx,vy,vz);
```

Now we define two discs, which will be intersected with the surface.

```
>cl=povdisc([0.5,0.5,0],[1,1,0],2); ...
```

```
> ll=povdisc([0,0,1/4],[0,0,1],2);
```

Write the surface minus the two discs.

```
>writeln(povdifference(mesh,povunion([cl,ll]),povlook(green)));
```

Write the two intersections.

```
>writeln(povintersection([mesh,cl],povlook(red))); ...
```

```
> writeln(povintersection([mesh,ll],povlook(gray)));
```

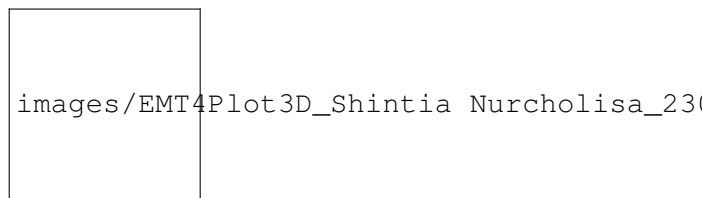
Write a point at the maximum.

```
>writeln(povpoint([1/2,1/2,1/4],povlook(gray),size=2*defaultpointsize));
```

Add axes and finish.

```
>writeAxes(0,1,0,1,0,1,d=0.015); ...
```

```
> povend();
```



Gambar 12.223 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-082.png

ANAGLYPHS IN POVRAY

To generate an anaglyph for a red/cyan glasses, Povray must run twice from different camera positions. It generates two Povray files and two PNG files, which are loaded with the function `loadanaglyph()`.

Of course, you need red/cyan glasses to view the following examples properly.

The function `pov3d()` has a simple switch to generate anaglyphs.

```
>pov3d("-exp(-x2-y2)/2",r=2,height=45°,>anaglyph, ...  
> center=[0,0,0.5],zoom=3.5);
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.224 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-083.png

If you create a scene with objects, you need to put the generation of the scene into a function, and run it twice with different values for the `anaglyph` parameter.

```
>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
```

The function `povanaglyph()` does all this. The parameters are like in `povstart()` and `povend()` combined.

```
>povanaglyph("myscene",zoom=4.5);
```



images/EMT4Plot3D_Shintia Nurcholisa_23

Gambar 12.225 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-084.png

DEFINING OWN OBJECTS

The povray interface of Euler contains a lot of objects. But you are not restricted to these. You can create own objects, which combine other objects, or are completely new objects.

We demonstrate a torus. The Povray command for this is “torus”. So we return a string with this command and its parameters. Note that the torus is always centered at the origin.

```
>function povdonat (r1,r2,look=“”) ...
```

```
    return "torus {" +r1+", "+r2+look+"}";  
endfunction
```

Here is our first torus.

```
>t1=povdonat(0.8,0.2)
```

```
torus {0.8,0.2}
```

Let us use this object to create a second torus, translated and rotated.

```
>t2=povobject(t1,rotate=xrotate(90°),translate=[0.8,0,0])
```

```
object { torus {0.8,0.2}  
  rotate 90 *x  
  translate <0.8,0,0>  
}
```

Now we place these objects into a scene. For the look, we use Phong Shading.

```
>povstart(center=[0.4,0,0],angle=0°,zoom=3.8,aspect=1.5); ...  
> writeln(povobject(t1,povlook(green,phong=1))); ...  
> writeln(povobject(t2,povlook(green,phong=1))); ...  
>  
>povend();
```


calls the Povray program. However, in case of errors, it does not display the error. You should therefore use

```
>povend(<exit);
```

if anything did not work. This will leave the Povray window open.

```
>povend(h=320,w=480);
```



images/EMT4Plot3D_Shintia Nurcholisa_23030630071-

Gambar 12.226 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-085.png

Here is a more elaborate example. We solve

$$Ax \leq b, \quad x \geq 0, \quad c \cdot x \rightarrow \text{Max.}$$

and show the feasible points and the optimum in a 3D plot.

```
>A=[10,8,4;5,6,8;6,3,2;9,5,6];
```

```
>b=[10,10,10,10]';
```

```
>c=[1,1,1];
```

First, let us check, if this example has a solution at all.

```
>x=simplex(A,b,c,>max,>check)'
```

```
[0, 1, 0.5]
```

Yes, it has.

Next we define two objects. The first is the plane

$$a \cdot x \leq b$$

```
>function oneplane (a,b,look="" ) ...
```

```
    return povplane(a,b,look)
endfunction
```

Then we define the intersection of all half spaces and a cube.

```
>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
```

We can now plot the scene.

```
>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); ...
>
```

The following is a circle around the optimum.

```
>writeln(povintersection([povsphere(x,0.5),povplane(c,c.x')],
...
> povlook(red,0.9)));
```

And an error in the direction of the optimum.

```
>writeln(povarrow(x,c*0.5,povlook(red)));
```

We add text to the screen. Text is just a 3D object. We need to place and turn it according to our view.

```
>writeln(povtext("Linear Problem",[0,0.2,1.3],size=0.05,rotate=5°));
...
> povend();
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.227 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-088.png

Contoh

1. Fungsi Linear Dua Variabel

$$g(x, y) = x - 3y + 7$$

```
>function f(x,y):= x-3*y+7  
>plot3d("f"):
```



Gambar 12.228 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-090.png

- Fungsi $f(x,y)$ didefinisikan sebagai $x-3y+7$.
- Perintah “plot3d(“f”)” digunakan untuk memplot grafik 3D dari fungsi $f(x,y)$ menggunakan fungsi plot3d di EMT.
- Grafik yang dihasilkan akan menampilkan fungsi dalam tiga dimensi, dengan sumbu x dan y mewakili variabel masukan dan sumbu z mewakili nilai keluaran fungsi. Grafik akan menunjukkan bentuk fungsi dan perubahannya seiring dengan perubahan variabel masukan.

2. Fungsi Kuadrat Dua Variabel

$$f(x, y) = 2x^2 + 3y^2 + 20$$

```
>function f(x,y):= 2*x2+3*y2+20  
>plot3d("f"):
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.229 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-092.png

- Perintah “function f(x,y):= 2x^2+3y^2+20” berarti mendefinisikan fungsi matematika f(x,y) sebagai 2 kali x pangkat 2 ditambah 3 kali y pangkat 2 ditambah 20.
- Perintah “plot3d(“f”)” berarti membuat grafik tiga dimensi dari fungsi f(x,y) yang telah didefinisikan sebelumnya.

3. Fungsi Logaritma Dua Variabel

$$f(x, y) = \log(5xy)$$

```
>function f(x,y):= log((5*x)*y)
>plot3d(“f”):
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.230 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-094.png

- Input yang diberikan adalah fungsi matematika dua variabel, f(x,y), yang didefinisikan sebagai logaritma hasil kali 5x dan y.

- Perintah “plot3d(“f”)” digunakan untuk memplot grafik fungsi $f(x,y)$ dalam ruang tiga dimensi.

4. Fungsi Eksponen Dua Variabel

$$f(x, y) = y^{x+9}$$

```
>function f(x,y):= y^(x+9)
>plot3d(“f”):
```



Gambar 12.231 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-096.png

- Perintah fungsi $f(x, y) := y^{(x+9)}$ adalah fungsi matematika dua variabel x dan y dan dengan rumus $y^{(x+9)}$
- Perintah `plot3d("f")` digunakan untuk memplot fungsi dalam ruang tiga dimensi. Plot yang dihasilkan akan menampilkan nilai fungsi sebagai permukaan pada bidang x - y , dengan tinggi permukaan mewakili nilai fungsi pada titik tersebut.

5. Fungsi Trigonometri Dua Variabel

$$f(x, y) = \cos(10x) - \sin(20y)$$

```
>function f(x,y):= cos(10*x)-sin(20*y)
>plot3d(“f”):
```



images/EMT4Plot3D_Shintia Nurcholis

Gambar 12.232 images/EMT4Plot3D_Shintia%20Nurcholisa_23030630071-098.png

- Perintah “function f(x,y):= cos(10x)-sin(20y)” adalah perintah untuk mendefinisikan fungsi matematika f(x,y) yang menghasilkan nilai cosinus dari x dikalikan dengan sinus dari y.
- Perintah “plot3d(”f”)” adalah perintah untuk membuat grafik tiga dimensi dari fungsi f(x,y) yang telah didefinisikan sebelumnya.

unicodehyperref hyphensurl []book xcolor amsmath,amssymb
iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype
[protrusion]basicmath parskip graphicx bookmark xurl

KALKULUS DENGAN EMT

Materi Kalkulus mencakup di antaranya:

- Fungsi (fungsi aljabar, trigonometri, eksponensial, logaritma, komposisi fungsi) Limit Fungsi, Turunan Fungsi, Integral Tak Tentu, Integral Tentu dan Aplikasinya, Barisan dan Deret (kekonvergenan barisan dan deret).

EMT (bersama Maxima) dapat digunakan untuk melakukan semua perhitungan di dalam kalkulus, baik secara numerik maupun analitik (eksak).

Mendefinisikan Fungsi

Terdapat beberapa cara mendefinisikan fungsi pada EMT, yakni:

- Menggunakan format `nama_fungsi := rumus fungsi` (untuk fungsi numerik),
- Menggunakan format `nama_fungsi &= rumus fungsi` (untuk fungsi simbolik, namun dapat dihitung secara numerik),
- Menggunakan format `nama_fungsi &&= rumus fungsi` (untuk fungsi simbolik murni, tidak dapat dihitung langsung),
- Fungsi sebagai program EMT.

Setiap format harus diawali dengan perintah `function` (bukan sebagai ekspresi).

Berikut adalah beberapa contoh cara mendefinisikan fungsi:

$$f(x) = 2x^2 + e^{\sin(x)}.$$

```
>function f(x) := 2*x^2+exp(sin(x)) // fungsi numerik
>f(0), f(1), f(pi)
```

```
1
4.31977682472
20.7392088022
```

```
>f(a) // tidak dapat dihitung nilainya
```

```
Variable or function a not found.
```

```
Error in:
```

```
f(a) // tidak dapat dihitung nilainya ...
^
```

Silahkan anda plot kurva diatas

```
>plot2d("f(x)",-10,10):
```



images/EMT4Kalkulus_Shintia Nurcholisa_2

Gambar 12.233 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-002.png

Berikutnya kita definisikan fungsi:

$$g(x) = \frac{\sqrt{x^2 - 3x}}{x + 1}.$$

```
>function g(x) := sqrt(x^2-3*x)/(x+1)
```

```
>g(3)
```

```
0
```

```
>g(0)
```

```
0
```

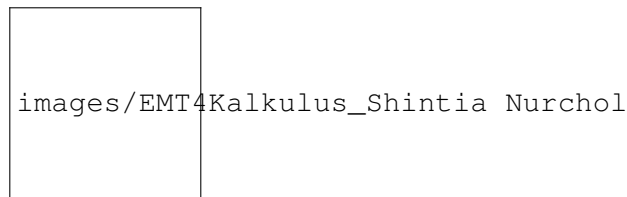
```
>g(1) // kompleks, tidak dapat dihitung oleh fungsi numerik
```

```

Floating point error!
Error in sqrt
Try "trace errors" to inspect local variables after e
g:
      useglobal; return sqrt(x^2-3*x)/(x+1)
Error in:
g(1) // kompleks, tidak dapat dihitung oleh fungsi nu
^

```

```
>plot2d("g(x)",-20,20,-20,20):
```



Gambar 12.234 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-004.png

Silakan Anda plot kurva fungsi di atas!

```
>f(g(5)) // komposisi fungsi
```

```
2.20920171961
```

```
>g(f(5))
```

```
0.950898070639
```

```
>function h(x) := f(g(x)) // definisi komposisi fungsi
```

```
>h(5) // sama dengan f(g(5))
```

```
2.20920171961
```

Silakan Anda plot kurva fungsi komposisi fungsi f dan g:

$$h(x) = f(g(x))$$

dan

$$u(x) = g(f(x))$$

bersama-sama kurva fungsi f dan g dalam satu bidang koordinat.

```
>plot2d("h(x)",-10,10,0,20):
```



Gambar 12.235 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-007.png

```
>function u(x):= g(f(x)) //definisi fungsi komposisi
```

```
>plot2d("u(x)",-5,5,0,5):
```



Gambar 12.236 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-008.png

```
>plot2d("f(x)",-20,20,-20,20); plot2d("g(x)",>add):
```



Gambar 12.237 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-009.png

```
>f(0:10) // nilai-nilai f(0), f(1), f(2), ..., f(10)
```

```
[1, 4.31978, 10.4826, 19.1516, 32.4692, 50.3833,  
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,  
99.929, 130.69, 163.51, 200.58]
```

```
>gmap(200:210)
```

```
[0.987534, 0.987596, 0.987657, 0.987718, 0.987778,  
0.987896, 0.987954, 0.988012, 0.988069, 0.988126]
```

Misalkan kita akan mendefinisikan fungsi

$$f(x) = \begin{cases} x^3 & x > 0 \\ x^2 & x \leq 0. \end{cases}$$

Fungsi tersebut tidak dapat didefinisikan sebagai fungsi numerik secara “inline” menggunakan format `:=`, melainkan didefinisikan sebagai program. Perhatikan, kata “map” digunakan agar fungsi dapat menerima vektor sebagai input, dan hasilnya berupa vektor. Jika tanpa kata “map” fungsinya hanya dapat menerima input satu nilai.

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

```
  if x>0 then return x^3  
  else return x^2  
endif;  
endfunction
```

```
>f(1)
```

```
1
```

```
>f(-2)
```



```
>f(-5:5)
```

```
[25, 16, 9, 4, 1, 0, 1, 8, 27, 64, 125]
```

```
>aspect(1.5); plot2d("f(x)",-5,5):
```



Gambar 12.238 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-011.png

```
>function f(x) &= 2*E^x // fungsi simbolik
```

$$2^x$$

```
>$f(a) // nilai fungsi secara simbolik
```

$$2e^a$$

```
>f(E) // nilai fungsi berupa bilangan desimal
```

```
30.308524483
```

```
>$f(E), $float(%)
```

$$2e^e$$

```
30.30852448295852
```

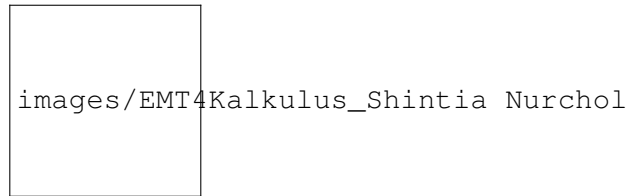
```
>function g(x) &= 3*x+1
```

$$3x + 1$$

```
>function h(x) &= f(g(x)) // komposisi fungsi
```

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

```
>plot2d("h(x)",-1,1):
```



Gambar 12.239 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-015.png

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.

Latihan soal:

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

```
>function f(x):=x^2+5;...
```

```
> function g(x):=x^3-5
```

```
>f(10), g(10)
```

```
105
```

```
995
```

Akan mencari komposisi fungsi

$$(f \circ g)(10)$$

$$(g \circ f)(10)$$

```
>function u(x):= f(g(x));...
```

```
> function v(x):= g(f(x))
```

```
>u(10), v(10)
```

```
990030
```

```
1157620
```

Akan digambarkan plotnya

```
>plot2d("u(x)");
```

```
>plot2d("v(x)");
```



Gambar 12.240 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-019.png



Gambar 12.241 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-020.png

MENGHITUNG LIMIT

Perhitungan limit pada EMT dapat dilakukan dengan menggunakan fungsi Maxima, yakni “limit”. Fungsi “limit” dapat digunakan untuk menghitung limit fungsi dalam bentuk ekspresi maupun fungsi yang sudah didefinisikan sebelumnya. Nilai limit dapat dihitung pada sebarang nilai atau pada tak hingga (-inf, minf, dan inf). Limit kiri dan limit kanan juga dapat dihitung, dengan cara memberi opsi “plus” atau “minus”. Hasil limit dapat berupa nilai, “und” (tak definisi), “ind” (tak tentu namun terbatas), “infinity” (kompleks tak hingga).

Perhatikan beberapa contoh berikut. Perhatikan cara menampilkan perhitungan secara lengkap, tidak hanya menampilkan hasilnya saja.

```
>$showev('limit(sqrt(x^2-3*x)/(x+1),x,inf))
```

$$\lim_{x \rightarrow \infty} \frac{\sqrt{x^2 - 3x}}{x + 1} = 1$$

```
>$limit((x^3-13*x^2+51*x-63)/(x^3-4*x^2-3*x+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^3-13*x^2+51*x-63)/(x^3-4*x^2-3*x+18)",0,4);
plot2d(3,-4/5,>points,style="ow",>add):
```

```
>$limit(2*x*sin(x)/(1-cos(x)),x,0)
```

$$4$$

$$2 \left(\lim_{x \rightarrow 0} \frac{x \sin x}{1 - \cos x} \right) = 4$$

Fungsi tersebut diskontinu di titik x=0. Berikut adalah grafik fungsinya.



Gambar 12.242 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-024.png



Gambar 12.243 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-027.png

```
>plot2d("2*x*sin(x)/(1-cos(x))",-pi,pi); plot2d(0,4,>points,style="ow",>add);
>$limit(cot(7*h)/cot(5*h),h,0)
```

$$\lim_{h \rightarrow 0} \frac{\cot(7h)}{\cot(5h)} = \frac{5}{7}$$

Fungsi tersebut juga diskontinu (karena tidak terdefinisi) di $x=0$. Berikut adalah grafiknya.

```
>plot2d("cot(7*x)/cot(5*x)",-0.001,0.001); plot2d(0,5/7,>points,style="ow",>add);
>$showev('limit(((x/8)^(1/3)-1)/(x-8),x,8))
```

$$\lim_{x \rightarrow 8} \frac{\frac{x^{\frac{1}{3}}}{2} - 1}{x - 8} = \frac{1}{24}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>aspet(1.5); plot2d("((x/8)^(1/3)-1)/(x-8)",-10,10); plot2d(8,1/24,>points,style="ow",>add);
```



Gambar 12.244 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-030.png



Gambar 12.245 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-032.png

```
>$showev('limit(1/(2*x-1),x,0))
```

$$\lim_{x \rightarrow 0} \frac{1}{2x-1} = -1$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>aspect(1.5); plot2d("1/(2x-1)",-10,10,-10,10); plot2d(0,-1,>points,style="ow",>add):
```



Gambar 12.246 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-034.png

```
>$showev('limit((x^2-3*x-10)/(x-5),x,5))
```


$$\lim_{x \rightarrow 5} \frac{x^2 - 3x - 10}{x - 5} = 7$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>aspect(1.5);      plot2d("(x^2-3*x-10)/(x-5)",-10,10,-10,10);
plot2d(5,7,>points,style="ow",>add):
```



Gambar 12.247 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-036.png

```
>$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.

```
>aspect(1.5); plot2d("sqrt(x^2+x)-x"); plot2d(10000,0.5,>points,style="ow",>add):
```



Gambar 12.248 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-038.png

```
>$showev('limit(abs(x-1)/(x-1),x,1,minus))
```

$$\lim_{x \uparrow 1} \frac{|x - 1|}{x - 1} = -1$$

Hitung limit di atas untuk x menuju 1 dari kanan.

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>plot2d("abs(x-1)/(x-1)",0,1,-2,2);      plot2d("abs(x-1)/(x-1)",1,2,-2,2,>add); plot2d(1,-1,>points,>add); plot2d(1,1,>points,>add):
```



images/EMT4Kalkulus_Shintia Nurchol

Gambar 12.249 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-040.png

```
>$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/EMT4Kalkulus_Shintia Nurchol

Gambar 12.250 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-042.png

```
>$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.

```
>aspect(1.5); plot2d("(sin(x^3))/x",-2*pi,2*pi); plot2d(0,0,>points,style=
```



Gambar 12.251 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-044.png

```
>$showev('limit(log(x), x, minf))
```

$$\lim_{x \rightarrow -\infty} \log x = \textit{infinity}$$

```
>$showev('limit((-2)^x,x, inf))
```

$$\lim_{x \rightarrow \infty} (-2)^x = \textit{infinity}$$

```
>$showev('limit(t-sqrt(2-t),t,2,minus))
```

$$\lim_{t \uparrow 2} t - \sqrt{2-t} = 2$$

```
>$showev('limit(t-sqrt(2-t),t,2,plus))
```

$$\lim_{t \downarrow 2} t - \sqrt{2-t} = 2$$

```
>$showev('limit(t-sqrt(2-t),t,5,plus)) // Perhatikan hasilnya
```

$$\lim_{t \downarrow 5} t - \sqrt{2-t} = 5 - \sqrt{3}i$$

```
>plot2d("x-sqrt(2-x)",0,2):
```

```
>$showev('limit((x^(2-9)/(2*x^2-5*x-3),x,3))
```

$$\lim_{x \rightarrow 3} \frac{x^2 - 9}{2x^2 - 5x - 3} = \frac{6}{7}$$

Tunjukkan limit tersebut dengan grafik, seperti contoh-contoh sebelumnya.

```
>aspect(1.5); plot2d("(x^(2-9)/(2*x^2-5*x-3))",-5,5,-10,10); plot2d(3,6/7,>points,styl
```



Gambar 12.252 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-050.png



Gambar 12.253 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-052.png

```
>$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.

```
>aspect(1.5); plot2d("(1-cos(x))/x",-10*pi,10*pi); plot2d(0,0,>points,sty
```



Gambar 12.254 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-054.png

```
>$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.

```
> aspect(1.5);    plot2d("(x2+abs(x)"/(x2-abs(x)))",-20,20,-20,20);
plot2d(0,-1,>points,style="ow",>add):
```



Gambar 12.255 images/EMT4Kalkulus_Shintia%20Nurcholisa.23030630071-056.png

```
>$showev('limit((1+1/x)^x,x,inf))
```

$$\lim_{x \rightarrow \infty} \left(\frac{1}{x} + 1 \right)^x = e$$

```
>plot2d("(1+1/x)^x",0,1000):
```



Gambar 12.256 images/EMT4Kalkulus_Shintia%20Nurcholisa.23030630071-058.png

```
>$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.

```
>aspect(1.5); plot2d("(1+x)^(1/x)",-20,20,-20,20); plot2d(0,E,>points,sty
```



Gambar 12.257 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-061.png

```
>$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.

```
>aspect(1.5); plot2d("(E^x-E^2)/(x-2)",-10,10,-10,10); plot2d(2,E^2,>points
```



Gambar 12.258 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-064.png

```
>$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)",-0.001,0.001):
```



Gambar 12.259 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-067.png

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.

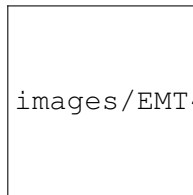
1. Hitung dan sketsakan limit di bawah ini

$$\lim_{x \rightarrow 0} \frac{x^3 - 1}{x - 1}$$

```
>$showev('limit((x^3-1)/(x-1),x,0))
```

$$\lim_{x \rightarrow 0} \frac{x^3 - 1}{x - 1} = 1$$

```
>aspect(1.5); plot2d("(x^3-1)/(x-1)",-10,10,-10,10); plot2d(0,1,>points,style="o")
```



images/EMT4Kalkulus_Shintia Nurcholisa

Gambar 12.260 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-070.png

2. Hitung dan sketsakan limit di bawah ini

$$\lim_{x \rightarrow 0} \frac{\sin(x)}{\cos(4x)}$$

```
>$showev('limit(sin(x)/cos(4*x),x,0))
```


$$\lim_{x \rightarrow 0} \frac{\sin x}{\cos(4x)} = 0$$

```
>aspect(1.5); plot2d("(sin(x))/(cos(4*x))",-20,20,-10,10); plot2d(0,0,>points,style="o"
```



Gambar 12.261 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-073.png

TURUNAN FUNGSI

Definisi turunan:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

Berikut adalah contoh-contoh menentukan turunan fungsi dengan menggunakan definisi turunan (limit).

>\$showev('limit(((x+h)^2-x^2)/h,h,0)) // turunan x^2

$$\lim_{h \rightarrow 0} \frac{(x+h)^2 - x^2}{h} = 2x$$

>p &= expand((x+h)^2-x^2)|simplify; \$p //pembilang dijabarkan dan disederhanakan

$$2hx + h^2$$

>q &=ratsimp(p/h); \$q // ekspresi yang akan dihitung limitnya disederhanakan

$$2x + h$$

>\$limit(q,h,0) // nilai limit sebagai turunan

$$2x$$

>\$showev('limit(((x+h)^n-x^n)/h,h,0)) // turunan x^n

$$\lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h} = nx^{n-1}$$

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.

Bukti

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

Untuk

$$f(x) = x^n$$

$$\frac{d}{dx} \sin(x) = \lim_{h \rightarrow 0} \frac{(x+h)^n - x^n}{h}$$

Dengan

$$(a+b)^n = \sum_{k=0}^n a^k b^{n-k}$$

maka

$$\begin{aligned} &= \lim_{h \rightarrow 0} \frac{(x^n + \frac{n}{1!}x^{n-1}h + \frac{n(n-1)}{2!}x^{n-2}h^2 + \frac{n(n-1)(n-2)}{3!}x^{n-3}h^3 + \dots) - x^n}{h} \\ &= \lim_{h \rightarrow 0} \frac{n.x^{n-1}h + \frac{n(n-1)}{2!}x^{n-2}h^2 + \frac{n(n-1)(n-2)}{3!}x^{n-3}h^3 + \dots}{h} \\ &= \lim_{h \rightarrow 0} n.x^{n-1} + \frac{n(n-1)}{2!}.x^{n-2}h + \frac{n(n-1)(n-2)}{3!}.x^{n-3}h^2 + \dots \\ &= n.x^{n-1} + 0 + 0 + \dots + 0 \\ &= n.x^{n-1} \end{aligned}$$

Jadi, terbukti benar bahwa

$$f'(x^n) = n.x^{n-1}$$

>\$showev('limit((sin(x+h)-sin(x))/h,h,0)) // turunan sin(x)

$$\lim_{h \rightarrow 0} \frac{\sin(x+h) - \sin x}{h} = \cos x$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut

benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, ekspansikan $\sin(x+h)$ dengan menggunakan rumus jumlah dua sudut.

Bukti

$$\begin{aligned}
 f'(x) &= \lim_{h \rightarrow 0} \frac{\sin(x+h) - \sin(x)}{h} \\
 \sin(a+b) &= \sin(a)\cos(b) + \cos(a)\sin(b) \\
 &= \lim_{h \rightarrow 0} \frac{\sin(x)\cos(h) + \cos(x)\sin(h) - \sin(x)}{h} \\
 &= \lim_{h \rightarrow 0} \sin(x) \cdot \frac{\cos(h) - 1}{h} + \lim_{h \rightarrow 0} \cos(x) \cdot \frac{\sin(h)}{h} \\
 &= \sin(x) \cdot 0 + \cos(x) \cdot 1 \\
 &= \cos(x)
 \end{aligned}$$

Jadi, terbukti benar bahwa

$$f'(\sin(x)) = \cos(x)$$

—

>\$showev('limit((log(x+h)-log(x))/h,h,0)) // turunan log(x)

$$\lim_{h \rightarrow 0} \frac{\log(x+h) - \log x}{h} = \frac{1}{x}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut

benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Sebagai petunjuk, gunakan sifat-sifat logaritma dan hasil limit pada bagian sebelumnya di atas.

Bukti

$$\begin{aligned}
 f'(x) &= \lim_{h \rightarrow 0} \frac{\log(x+h) - \log x}{h} \\
 &= \lim_{h \rightarrow 0} \frac{\frac{d}{dh}(\log(x+h) - \log x)}{\frac{d}{dh}(h)} \\
 &= \lim_{h \rightarrow 0} \frac{\frac{1}{x+h}}{1} \\
 &= \lim_{h \rightarrow 0} \frac{1}{x+h} \\
 &= \frac{1}{x}
 \end{aligned}$$

Jadi, terbukti benar bahwa

$$f'(x) = \lim_{h \rightarrow 0} \frac{\log(x+h) - \log x}{h} = \frac{1}{x}$$

```
>$showev('limit((1/(x+h)-1/x)/h,h,0)) // turunan 1/x
```

$$\lim_{h \rightarrow 0} \frac{\frac{1}{x+h} - \frac{1}{x}}{h} = -\frac{1}{x^2}$$

```
>$showev('limit((E^(x+h)-E^x)/h,h,0)) // turunan f(x)=e^x
```

```
Answering "Is x an integer?" with "integer"
```

```
Answering "Is x an integer?" with "integer"
```

```
Answering "Is x an integer?" with "integer"
```

```
Answering "Is x an integer?" with "integer"
```

```
Answering "Is x an integer?" with "integer"
```

```
Maxima is asking
```

```
Acceptable answers are: yes, y, Y, no, n, N, unknown, uk
```

```
Is x an integer?
```

```
Use assume!
```

```
Error in:
```

```
$showev('limit((E^(x+h)-E^x)/h,h,0)) // turunan f(x)=e^x .
```

Maxima bermasalah dengan limit:

$$\lim_{h \rightarrow 0} \frac{e^{x+h} - e^x}{h}.$$

Oleh karena itu diperlukan trik khusus agar hasilnya benar.

```
>$showev('limit((E^h-1)/h,h,0))
```

$$\lim_{h \rightarrow 0} \frac{e^h - 1}{h} = 1$$

```
>$showev('factor(E^(x+h)-E_x))
```

$$factor(e^{x+h} - e^x) = (e^h - 1) e^x$$

```
>$showev('limit(factor((E^(x+h)-E_x)/h),h,0)) // turunan f(x)=e^x
```

$$\left(\lim_{h \rightarrow 0} \frac{e^h - 1}{h} \right) e^x = e^x$$

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

x
x

```
>$showev('limit(f(x),x,0))
```

$$\lim_{x \rightarrow 0} x^x = 1$$

Silakan Anda gambar kurva

$$y = x^x.$$

```
>aspect(1.5); plot2d("x^x"); plot2d(0,1,>points,style="ow",>add):
```

```
>$showev('limit((f(x+h)-f(x))/h,h,0)) // turunan f(x)=x^x
```

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h} = infinity$$

Di sini Maxima juga bermasalah terkait limit:



Gambar 12.262 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-112.png

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h}.$$

Dalam hal ini diperlukan asumsi nilai x .

```
>&assume(x>0);    $showev('limit((f(x+h)-f(x))/h,h,0)) //
turunan f(x)=x^x
```

$$\lim_{h \rightarrow 0} \frac{(x+h)^{x+h} - x^x}{h} = x^x (\log x + 1)$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

```
>&forget(x>0) // jangan lupa, lupakan asumsi untuk kembali
ke semula
```

```
[x > 0]
```

```
>&forget(x<0)
```

```
[x < 0]
```

```
>&facts()
```

```
[]
```

```
>$showev('limit((asin(x+h)-asin(x))/h,h,0)) // turunan arcsin(x)
```

$$\lim_{h \rightarrow 0} \frac{\arcsin(x+h) - \arcsin(x)}{h} = \frac{1}{\sqrt{1-x^2}}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

>\$showev('limit((tan(x+h)-tan(x))/h,h,0)) // turunan tan(x)

$$\lim_{h \rightarrow 0} \frac{\tan(x+h) - \tan x}{h} = \frac{1}{\cos^2 x}$$

Mengapa hasilnya seperti itu? Tuliskan atau tunjukkan bahwa hasil limit tersebut benar, sehingga benar turunan fungsinya benar. Tulis penjelasan Anda di komentar ini.

Bukti

Gunakan identitas trigonometri untuk menulis selisih

$$\tan(x+h), \tan(x)$$

Kita tahu bahwa ada identitas trigonometri untuk selisih dua sudut pada fungsi tangen:

$$\tan(A) - \tan(B) = \frac{\sin(A-B)}{\cos(A)\cos(B)}$$

dan

$$\tan(x) = \frac{\sin(x)}{\cos(x)}$$

Jadi, selisih untuk

$$\begin{aligned} & \tan(x+h) - \tan(x) : \\ & \tan(x+h) - \tan(x) = \frac{\sin(x+h)}{\cos(x+h)} - \frac{\sin(x)}{\cos(x)} \\ & = \frac{\sin(x+h) \cdot \cos(x) - \sin(x) \cdot \cos(x+h)}{\cos(x+h) \cdot \cos(x)} \\ & = \frac{\sin((x+h) - x)}{\cos(x+h) \cdot \cos(x)} \end{aligned}$$

$$= \frac{\sin(h)}{\cos(x+h) \cdot \cos(x)}$$

Sehingga nilai limit tersebut menjadi

$$\begin{aligned} \lim_{h \rightarrow 0} \frac{\tan(x+h) - \tan(x)}{h} &= \lim_{h \rightarrow 0} \frac{\frac{\sin(h)}{\cos(x+h) \cdot \cos(x)}}{h} \\ &= \lim_{h \rightarrow 0} \frac{\sin(h)}{h} \cdot \lim_{h \rightarrow 0} \frac{1}{\cos(x+h) \cdot \cos(x)} \end{aligned}$$

Kita tahu bahwa

$$\lim_{h \rightarrow 0} \frac{\sin(h)}{h} = 1$$

dan juga

$$\lim_{h \rightarrow 0} \cos(x+h) = \cos(x)$$

Jadi,

$$\begin{aligned} \lim_{h \rightarrow 0} \frac{\tan(x+h) - \tan(x)}{h} &= 1 \cdot \lim_{h \rightarrow 0} \frac{1}{\cos(x) \cdot \cos(x)} \\ &= \lim_{h \rightarrow 0} \frac{1}{\cos^2(x)} \\ &= \frac{1}{\cos^2(x)} \end{aligned}$$

—

```
>function f(x) &= sinh(x) // definisikan f(x)=sinh(x)
```

```
sinh(x)
```

```
>function df(x) &= limit((f(x+h)-f(x))/h,h,0); $df(x) // df(x) =  
f'(x)
```

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

Hasilnya adalah $\cosh(x)$, karena

$$\frac{e^x + e^{-x}}{2} = \cosh(x).$$

```
>plot2d(["f(x)","df(x)"],-pi,pi,color=[blue,red]):
```



images/EMT4Kalkulus_Shintia Nurchol

Gambar 12.263 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-135.png

```
>function f(x) &= sin(3*x5+7)2
```

$$\sin(3x^5 + 7)$$

```
>diff(f,3), diffc(f,3)
```

```
1198.32948904
```

```
1198.72863721
```

Apakah perbedaan diff dan diffc?

```
>$showev('diff(f(x),x))
```

$$\frac{d}{dx} \sin^2(3x^5 + 7) = 30x^4 \cos(3x^5 + 7) \sin(3x^5 + 7)$$

```
>$% with x=3
```

$$\%at \left(\frac{d}{dx} \sin^2(3x^5 + 7), x = 3 \right) = 2430 \cos 736 \sin 736$$

```
>$float(%)
```

$$\%at \left(\frac{d^{1.0}}{dx^{1.0}} \sin^2 (3.0 x^5 + 7.0) , x = 3.0 \right) = 1198.728637211748$$

>plot2d(f,0,3.1):



Gambar 12.264 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-139.png

>function f(x) &=5*cos(2*x)-2*x*sin(2*x) // 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)

$$f(1) = 5 \cos 2 - 2 \sin 2$$

$$-3.899329036387075$$

$$f(2) = 5 \cos 4 - 4 \sin 4$$

$$-0.2410081230863468$$

>xp=solve("df(x)",1,2,0) // solusi f'(x)=0 pada interval [1, 2]

$$1.35822987384$$

>df(xp), f(xp) // cek bahwa f'(xp)=0 dan nilai ekstrim di titik tersebut

0

-5.67530133759

```
>plot2d(["f(x)","df(x)"],0,2*pi,color=[blue,red]): //grafik  
fungsi dan turunannya
```



Gambar 12.265 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-144.png

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.

1. Tentukan nilai turunan berikut dan sketsakan grafiknya

$$f(x) = x^3 + 3x^2 + 2x + 1$$

```
>function a(x) &= x3+3*x2+2*x+1; $a(x)
```

$$x^3 + 3x^2 + 2x + 1$$

```
>function da(x) &= limit((a(x+h)-a(x))/h,h,0); &da(x) // da(x)=a'(x)
```

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

```
>plot2d(["a(x)", "da(x)"], -10, 10, -10, 20, color=[blue, green]):
```



images/EMT4Kalkulus_Shintia Nurchol

Gambar 12.266 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-147.png

2. Tentukan nilai turunan berikut dan sketsakan grafiknya

$$f(x) = \frac{1}{3x}$$

```
>function b(x) &= 1/(3*x); $b(x)
```

$$\frac{1}{3x}$$

```
>function db(x) &= limit((b(x+h)-b(x))/h,h,0); $ db(x) // db(x)=b'(x)
```

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

```
>plot2d(["b(x)","db(x)"],-10,10,-10,10,color=[blue,red]):
```



images/EMT4Kalkulus_Shintia Nurcholisa_23030630071-151.png

Gambar 12.267 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-151.png

3. Tentukan nilai turunan berikut dan sketsakan grafiknya

$$f(x) = \ln(x)$$

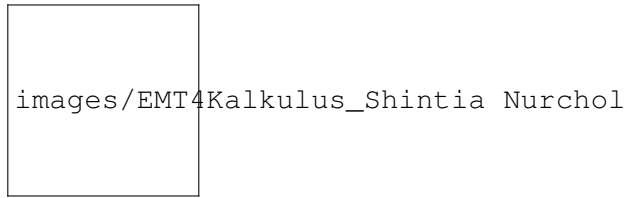
```
>function c(x) &= ln(x); $c(x)
```

$$\log x$$

```
>function dc(x) &= limit((c(x+h)-c(x))/h,h,0); $dc(x) // dc(x)=c'(x)
```

$$\frac{1}{x}$$

```
>plot2d(["c(x)","dc(x)"],-10,10,-10,10,color=[green,yellow]):
```



Gambar 12.268 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-155.png

4. Tentukan nilai turunan berikut dan sketsakan grafiknya

$$f(x) = \tan(x)$$

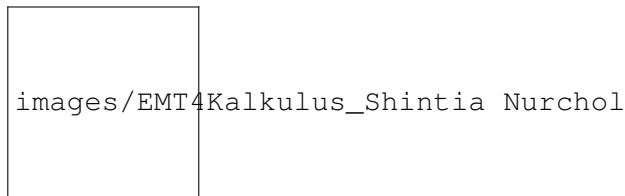
```
>function d(x) &= tan(x); $d(x)
```

$$\tan x$$

```
>function dd(x) &= limit((d(x+h)-d(x))/h,h,0); $dd(x) //dd(x)=d'(x)
```

$$\frac{1}{\cos^2 x}$$

```
>plot2d(["d(x)","dd(x)"],-10,10,-5,5,color=[yellow,black]):
```



Gambar 12.269 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-159.png

5. Tentukan nilai turunan berikut dan sketsakan grafiknya

$$f(x) = (4x^2 + 6x)^{10}$$


```
>function e(x) &= (4*x^2+6*x)^10; $e(x)
```

$$(4x^2 + 6x)^{10}$$

```
>function de(x) &= diff(e(x),x); $de(x) //turunan rantai
```

$$10(8x + 6)(4x^2 + 6x)^9$$

```
>plot2d(["e(x)","de(x)"],-10,0,-10,20,color=[green,red]):
```



images/EMT4Kalkulus_Shintia Nurcholisa_2

Gambar 12.270 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-163.png

INTEGRAL

EMT dapat digunakan untuk menghitung integral, baik integral tak tentu maupun integral tentu. Untuk integral tak tentu (simbolik) sudah tentu EMT menggunakan Maxima, sedangkan untuk perhitungan integral tentu EMT sudah menyediakan beberapa fungsi yang mengimplementasikan algoritma kuadratur (perhitungan integral tentu menggunakan metode numerik).

Pada notebook ini akan ditunjukkan perhitungan integral tentu dengan menggunakan Teorema Dasar Kalkulus:

$$\int_a^b f(x) dx = F(b) - F(a), \quad \text{dengan } F'(x) = f(x).$$

Fungsi untuk menentukan integral adalah `integrate`. Fungsi ini dapat digunakan untuk menentukan, baik integral tentu maupun tak tentu (jika fungsinya memiliki antiderivatif). Untuk perhitungan integral tentu fungsi `integrate` menggunakan metode numerik (kecuali fungsinya tidak integrabel, kita tidak akan menggunakan metode ini).

```
>$showev('integrate(x^n,x))
```

Answering "Is n equal to -1?" with "no"

$$\int x^n dx = \frac{x^{n+1}}{n+1}$$

```
>$showev('integrate(1/(1+x),x))
```

$$\int \frac{1}{x+1} dx = \log(x+1)$$

```
>$showev('integrate(1/(1+x^2),x))
```

$$\int \frac{1}{x^2+1} dx = \arctan x$$

```
>$showev('integrate(1/sqrt(1-x^2),x))
```

$$\int \frac{1}{\sqrt{1-x^2}} dx = \arcsin x$$

```
>$showev('integrate(sin(x),x,0,pi))
```

$$\int_0^\pi \sin x \, dx = 2$$

```
>plot2d("sin(x)",0,2*pi):
```



Gambar 12.271 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-170.png

```
>$showev('integrate(sin(x),x,a,b))
```

$$\int_a^b \sin x \, dx = \cos a - \cos b$$

```
>$showev('integrate(x^n,x,a,b))
```

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

$$\int_a^b x^n \, dx = \frac{b^{n+1}}{n+1} - \frac{a^{n+1}}{n+1}$$

```
>$showev('integrate(x^2*sqrt(2*x+1),x))
```

$$\int x^2 \sqrt{2x+1} \, dx = \frac{(2x+1)^{\frac{7}{2}}}{28} - \frac{(2x+1)^{\frac{5}{2}}}{10} + \frac{(2x+1)^{\frac{3}{2}}}{12}$$

```
>$showev('integrate(x^2*sqrt(2*x+1),x,0,2))
```

$$\int_0^2 x^2 \sqrt{2x+1} \, dx = \frac{2 \cdot 5^{\frac{5}{2}}}{21} - \frac{2}{105}$$

>\$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)

$$\frac{e^{-x/2}}{E}$$

>\$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):



Gambar 12.272 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-181.png

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 $x = 0.1, 0.2, 0.3, \dots, 3.2$.

>0.1*sum(f(x+0.1)) // cek langsung dengan perhitungan numerik EMT

0.836219610253

Untuk mendapatkan nilai integral tentu yang mendekati nilai sebenarnya, lebar sub-intervalnya dapat diperkecil lagi, sehingga daerah di bawah kurva tertutup semuanya, misalnya dapat digunakan lebar subinterval 0.001. (Silakan dicoba!)

Meskipun Maxima tidak dapat menghitung integral tentu fungsi tersebut untuk batas-batas yang berhingga, namun integral tersebut dapat dihitung secara eksak jika batas-batasnya tak hingga. Ini adalah salah satu keajaiban di dalam matematika, yang terbatas tidak dapat dihitung secara eksak, namun yang tak hingga malah dapat dihitung secara eksak.

```
>$showev('integrate(f(x),x,0,inf))
```

$$\int_0^{\infty} e^{-x^2} dx = \frac{\sqrt{\pi}}{2}$$

Tunjukkan kebenaran hasil di atas!

Berikut adalah contoh lain fungsi yang tidak memiliki antiderivatif, sehingga integral tentunya hanya dapat dihitung dengan metode numerik.

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

x
x

```
>$showev('integrate(f(x),x,0,1))
```

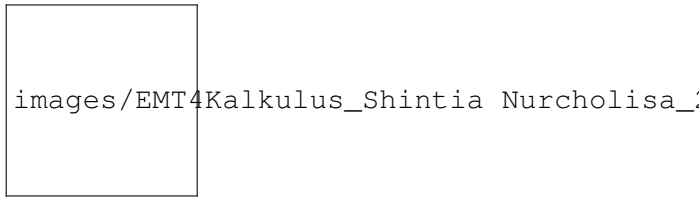
$$\int_0^1 x^x dx = \int_0^1 x^x dx$$

```
>x:=0:0.1:1-0.01; plot2d(x,f(x+0.01),>bar); plot2d("f(x)",0,1,>add):
```

Maxima gagal menghitung integral tentu tersebut secara langsung menggunakan perintah `integrate`. Berikut kita lakukan seperti contoh sebelumnya untuk mendapat hasil atau pendekatan nilai integral tentu tersebut.

```
>t &= makelist(a,a,0,1-0.01,0.01);
```

```
>fx &= makelist(f(t[i]+0.01),i,1,length(t));
```



Gambar 12.273 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-186.png

$$\int_0^1 x^x dx = 0.7834935879025506$$

Apakah hasil tersebut cukup baik? perhatikan gambarnya.

```
>function f(x) &= sin(3*x5+7)2
```

$$\sin(3x^2 + 7)$$

```
>integrate(f,0,1)
```

```
0.542581176074
```

```
>&showev('integrate(f(x),x,0,1))
```

$$\frac{\int_0^1 \sin(3x^2 + 7) dx}{\frac{\gamma(5) \sin(14) \sin(\frac{\pi}{10})}{10^{\frac{1}{5}}}} - \left(\left(6^{\frac{4}{5}} \gamma_{\text{incomplete}}\left(-, 6\right) + 6^{\frac{4}{5}} \gamma_{\text{incomplete}}\left(-, 6\right) \right) \sin(14) + \left(6^{\frac{4}{5}} \int_0^1 \gamma_{\text{incomplete}}\left(-, 6\right) \right) \right)$$

```

4/5      1      pi
- 6      I gamma_incomplete(-, - 6 I)) cos(14)) sin(--
5      10

>&float(%)

1.0
/
[      2      5
I      sin (3.0 x  + 7.0) dx =
]
/
0.0
0.09820784258795788 - 0.008333333333333333
(0.3090169943749474 (0.1367372182078336
(4.192962712629476 I gamma__incomplete(0.2, 6.0 I)
- 4.192962712629476 I gamma__incomplete(0.2, - 6.0 I)
+ 0.9906073556948704 (4.192962712629476 gamma__incomp
+ 4.192962712629476 gamma__incomplete(0.2, - 6.0 I))

>$showev('integrate(x*exp(-x),x,0,1)) // Integral tentu (eksak)

```

$$\int_0^1 x e^{-x} dx = 1 - 2e^{-1}$$

Aplikasi Integral Tentu

```

>plot2d("x^3-x",-0.1,1.1); plot2d("-x^2",>add); ...
> b=solve("x^3-x+x^2",0.5); x=linspace(0,b,200); xi=flipx(x); ...
> plot2d(x|xi,x^3-x|-xi^2,>filled,style="|",fillcolor=1,>add): // Plot
daerah antara 2 kurva

>a=solve("x^3-x+x^2",0), b=solve("x^3-x+x^2",1) // absis titik-titik
potong kedua kurva

```

```

0
0.61803398875

```

```

>integrate("(-x^2)-x(3-x)",a,b) // luas daerah yang diarsir

```

```

0.0758191713542

```




images/EMT4Kalkulus_Shintia Nurcholisa_2

Gambar 12.274 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-189.png

Hasil tersebut akan kita bandingkan dengan perhitungan secara analitik.

`>a = solve((-x2-(x3-x),x); $a // menentukan absis titik potong kedua kurva secara eksak`

$$\left[x = \frac{-\sqrt{5}-1}{2}, x = \frac{\sqrt{5}-1}{2}, x = 0 \right]$$

`>$showev('integrate(-x2-x3+x,x,0,(sqrt(5)-1)/2)) // Nilai integral secara eksak`

$$\int_0^{\frac{\sqrt{5}-1}{2}} -x^3 - x^2 + x \, dx = \frac{13 - 5^{\frac{3}{2}}}{24}$$

`>$float(%)`

$$\int_{0.0}^{0.6180339887498949} -1.0x^3 - 1.0x^2 + x \, dx = 0.07581917135421037$$

Panjang Kurva

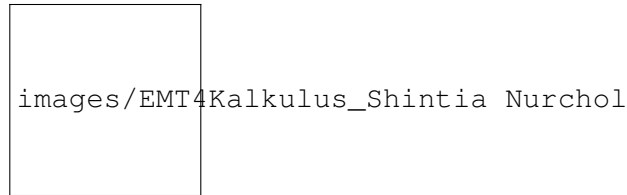
Hitunglah panjang kurva berikut ini dan luas daerah di dalam kurva tersebut.

$$\gamma(t) = (r(t) \cos(t), r(t) \sin(t))$$

dengan

$$r(t) = 1 + \frac{\sin(3t)}{2}, \quad 0 \leq t \leq 2\pi.$$

```
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t); y=r*sin(t); ...
> plot2d(x,y,>filled,fillcolor=red,style="P",r=1.5): // Kita gambar
kurvanya terlebih dahulu
```



Gambar 12.275 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-195.png

```
>function r(t) &= 1+sin(3*t)/2; $'r(t)=r(t)

r ([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2, 0.21, 0.22, 0.23, 0.24, 0.25, 0.26, 0.27, 0.28, 0.29, 0.3, 0.31, 0.32, 0.33, 0.34, 0.35, 0.36, 0.37, 0.38, 0.39, 0.4, 0.41, 0.42, 0.43, 0.44, 0.45, 0.46, 0.47, 0.48, 0.49, 0.5, 0.51, 0.52, 0.53, 0.54, 0.55, 0.56, 0.57, 0.58, 0.59, 0.6, 0.61, 0.62, 0.63, 0.64, 0.65, 0.66, 0.67, 0.68, 0.69, 0.7, 0.71, 0.72, 0.73, 0.74, 0.75, 0.76, 0.77, 0.78, 0.79, 0.8, 0.81, 0.82, 0.83, 0.84, 0.85, 0.86, 0.87, 0.88, 0.89, 0.9, 0.91, 0.92, 0.93, 0.94, 0.95, 0.96, 0.97, 0.98, 0.99, 1.0])

>function fx(t) &= r(t)*cos(t); $'fx(t)=fx(t)

fx ([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2, 0.21, 0.22, 0.23, 0.24, 0.25, 0.26, 0.27, 0.28, 0.29, 0.3, 0.31, 0.32, 0.33, 0.34, 0.35, 0.36, 0.37, 0.38, 0.39, 0.4, 0.41, 0.42, 0.43, 0.44, 0.45, 0.46, 0.47, 0.48, 0.49, 0.5, 0.51, 0.52, 0.53, 0.54, 0.55, 0.56, 0.57, 0.58, 0.59, 0.6, 0.61, 0.62, 0.63, 0.64, 0.65, 0.66, 0.67, 0.68, 0.69, 0.7, 0.71, 0.72, 0.73, 0.74, 0.75, 0.76, 0.77, 0.78, 0.79, 0.8, 0.81, 0.82, 0.83, 0.84, 0.85, 0.86, 0.87, 0.88, 0.89, 0.9, 0.91, 0.92, 0.93, 0.94, 0.95, 0.96, 0.97, 0.98, 0.99, 1.0])

>function fy(t) &= r(t)*sin(t); $'fy(t)=fy(t)

fy ([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2, 0.21, 0.22, 0.23, 0.24, 0.25, 0.26, 0.27, 0.28, 0.29, 0.3, 0.31, 0.32, 0.33, 0.34, 0.35, 0.36, 0.37, 0.38, 0.39, 0.4, 0.41, 0.42, 0.43, 0.44, 0.45, 0.46, 0.47, 0.48, 0.49, 0.5, 0.51, 0.52, 0.53, 0.54, 0.55, 0.56, 0.57, 0.58, 0.59, 0.6, 0.61, 0.62, 0.63, 0.64, 0.65, 0.66, 0.67, 0.68, 0.69, 0.7, 0.71, 0.72, 0.73, 0.74, 0.75, 0.76, 0.77, 0.78, 0.79, 0.8, 0.81, 0.82, 0.83, 0.84, 0.85, 0.86, 0.87, 0.88, 0.89, 0.9, 0.91, 0.92, 0.93, 0.94, 0.95, 0.96, 0.97, 0.98, 0.99, 1.0])

>function ds(t) &= trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2)));
$'ds(t)=ds(t)

Maxima said:
diff: second argument must be a variable; found errexpr
-- an error. To debug this try: debugmode(true);

Error in:
... e(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2)));
```

>\$integrate(ds(x),x,0,2*pi) //panjang (keliling) kurva

$$\int_0^{2\pi} ds(x) dx$$

Maxima gagal melakukan perhitungan eksak integral tersebut.

Berikut kita hitung integralnya secara umerik dengan perintah EMT.

>integrate(“ds(x)”,0,2*pi)

Function ds not found.

Try list ... to find functions!

Error in expression: ds(x)

%mapexpression1:

return expr(x,args());

Error in map.

%evalexpression:

if maps then return %mapexpression1(x,f\$;args());

gauss:

if maps then y=%evalexpression(f\$,a+h-(h*xn)',maps;args)

adaptivegauss:

t1=gauss(f\$,c,c+h;args(),=maps);

Try "trace errors" to inspect local variables after errors

integrate:

return adaptivegauss(f\$,a,b,eps*1000;args(),=maps);

Spiral Logaritmik

$$x = e^{ax} \cos x, \quad y = e^{ax} \sin x.$$

>a=0.1;plot2d(“exp(a*x)*cos(x)”,“exp(a*x)*sin(x)”,r=2,xmin=0,xmax=2*pi):

>&kill(a) // hapus expresi a

done

>function fx(t) &= exp(a*t)*cos(t); \$'fx(t)=fx(t)

fx ([0,0.01,0.02,0.03,0.04,0.05,0.06,0.07,0.08,0.09,0.1,0.11,0.12,0.13,0.14,0.1

Gambar 12.276 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-201.png

```
>function fy(t) &= exp(a*t)*sin(t); $'fy(t)=fy(t)
```

```
fy ([0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, 0.2, 0.21, 0.22, 0.23, 0.24, 0.25, 0.26, 0.27, 0.28, 0.29, 0.3, 0.31, 0.32, 0.33, 0.34, 0.35, 0.36, 0.37, 0.38, 0.39, 0.4, 0.41, 0.42, 0.43, 0.44, 0.45, 0.46, 0.47, 0.48, 0.49, 0.5, 0.51, 0.52, 0.53, 0.54, 0.55, 0.56, 0.57, 0.58, 0.59, 0.6, 0.61, 0.62, 0.63, 0.64, 0.65, 0.66, 0.67, 0.68, 0.69, 0.7, 0.71, 0.72, 0.73, 0.74, 0.75, 0.76, 0.77, 0.78, 0.79, 0.8, 0.81, 0.82, 0.83, 0.84, 0.85, 0.86, 0.87, 0.88, 0.89, 0.9, 0.91, 0.92, 0.93, 0.94, 0.95, 0.96, 0.97, 0.98, 0.99, 1.0])
```

```
>function df(t) &= trigreduce(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2)));
$'df(t)=df(t)
```

Maxima said:

```
diff: second argument must be a variable; found errexpr
-- an error. To debug this try: debugmode(true);
```

Error in:

```
... e(radcan(sqrt(diff(fx(t),t)^2+diff(fy(t),t)^2)));
```

```
>S &=integrate(df(t),t,0,2*pi); $S // panjang kurva (spiral)
```

Maxima said:

```
defint: variable of integration cannot be a constant;
-- an error. To debug this try: debugmode(true);
```

Error in:

```
S &=integrate(df(t),t,0,2*pi); $S // panjang kurva
```

```
>S(a=0.1) // Panjang kurva untuk a=0.1
```

Function S not found.

```
Try list ... to find functions!
Error in:
S(a=0.1) // Panjang kurva untuk a=0.1 ...
^
```

Soal:

Tunjukkan bahwa keliling lingkaran dengan jari-jari r adalah $K=2\pi r$.

Berikut adalah contoh menghitung panjang parabola.

```
>plot2d("x^2",xmin=-1,xmax=1):
```



images/EMT4Kalkulus_Shintia Nurcholisa_2

Gambar 12.277 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-204.png

```
>$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
```



images/EMT4Kalkulus_Shintia Nurchol

Gambar 12.278 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-207.png

Hasilnya mendekati panjang yang dihitung secara eksak. Untuk mendapatkan hampiran yang cukup akurat, jarak antar titik dapat diperkecil, misalnya 0.1, 0.05, 0.01, dan seterusnya. Cobalah Anda ulangi perhitungannya dengan nilai-nilai tersebut.

Koordinat Kartesius

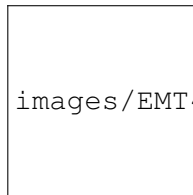
Berikut diberikan contoh perhitungan panjang kurva menggunakan koordinat Kartesius. Kita akan hitung panjang kurva dengan persamaan implisit:

$$x^3 + y^3 - 3xy = 0.$$

>z &= x^{3+y}3-3*x*y; \$z

$$y^3 - 3xy + x^3$$

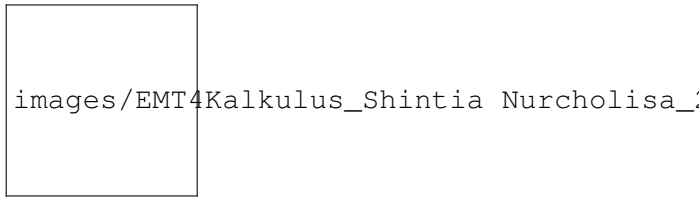
>plot2d(z,r=2,level=0,n=100):



images/EMT4Kalkulus_Shintia Nurchol

Gambar 12.279 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-210.png

Kita tertarik pada kurva di kuadran pertama.



Gambar 12.280 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-211.png

```
>plot2d(z,a=0,b=2,c=0,d=2,level=[-10;0],n=100,contourwidth=3,style="l");
```

Kita selesaikan persamaannya untuk x .

```
>$z with y=l*x, sol &= solve(%x); $sol
```

$$l^3 x^3 + x^3 - 3 l x^2$$

$$\left[x = \frac{3 l}{l^3 + 1}, x = 0 \right]$$

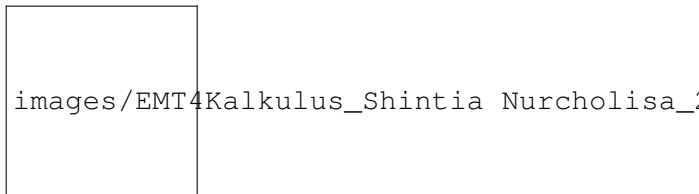
Kita gunakan solusi tersebut untuk mendefinisikan fungsi dengan Maxima.

```
>function f(l) &= rhs(sol[1]); $'f(l)=f(l)
```

$$f(l) = \frac{3 l}{l^3 + 1}$$

Fungsi tersebut juga dapat digunakan untuk menggambar kurvanya. Ingat, bahwa fungsi tersebut adalah nilai x dan nilai $y=lx$, yakni $x=f(l)$ dan $y=lf(l)$.

```
>plot2d(&f(x),&x*f(x),xmin=-0.5,xmax=2,a=0,b=2,c=0,d=2,r=1.5):
```



Gambar 12.281 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-215.png

Elemen panjang kurva adalah:

$$ds = \sqrt{f'(l)^2 + (lf'(l) + f(l))^2}.$$

```
>function ds(l) &= ratsimp(sqrt(diff(f(l),l)^2+diff(l*f(l),l)^2)); $'ds(l)=ds(l)
```

$$ds(l) = \frac{\sqrt{9l^8 + 36l^6 - 36l^5 - 36l^3 + 36l^2 + 9}}{\sqrt{l^{12} + 4l^9 + 6l^6 + 4l^3 + 1}}$$

```
>$integrate(ds(l),l,0,1)
```

$$\int_0^1 \frac{\sqrt{9l^8 + 36l^6 - 36l^5 - 36l^3 + 36l^2 + 9}}{\sqrt{l^{12} + 4l^9 + 6l^6 + 4l^3 + 1}} dl$$

Integral tersebut tidak dapat dihitung secara eksak menggunakan Maxima. Kita hitung integral tersebut secara numerik dengan Euler. Karena kurva simetris, kita hitung untuk nilai variabel integrasi dari 0 sampai 1, kemudian hasilnya dikalikan 2.

```
>2*integrate("ds(x)",0,1)
```

```
4.91748872168
```

>2*romberg(&ds(x),0,1)// perintah Euler lain untuk menghitung nilai hampiran integral yang 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)");
return romberg(ds,a,b);
endfunction
```



```
>panjangkurva("x","x^2",-1,1) // cek untuk menghitung
panjang kurva parabola sebelumnya
```

```
2.95788571509
```

Bandingkan dengan nilai eksak di atas.

```
>2*panjangkurva(mxm("f(x)",mxm("x*f(x)"),0,1) // cek
contoh terakhir, bandingkan hasilnya!
```

```
4.91748872168
```

Kita hitung panjang spiral Archimides berikut ini dengan fungsi tersebut.

```
>plot2d("x*cos(x)","x*sin(x)",xmin=0,xmax=2*pi,square=1):
```



Gambar 12.282 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-219.png

```
>panjangkurva("x*cos(x)","x*sin(x)",0,2*pi)
```

```
21.2562941482
```

Berikut kita definisikan fungsi yang sama namun dengan Maxima, untuk perhitungan eksak.

```
>&kill(ds,x,fx,fy)
```

```
done
```

```
>function ds(fx,fy) &&= sqrt(diff(fx,x)^2+diff(fy,x)^2)
```

```
sqrt(diff(fy, x)^2 + diff(fx, x)^2)
```

>sol &= ds(x*cos(x),x*sin(x)); \$sol // Kita gunakan untuk menghitung panjang kurva terakhir di atas

$$\sqrt{(\cos x - x \sin x)^2 + (\sin x + x \cos x)^2}$$

>\$sol | trigreduce | expand, \$integrate(%,x,0,2*pi), %()

$$\sqrt{x^2 + 1}$$

$$\frac{\operatorname{asinh}(2\pi) + 2\pi\sqrt{4\pi^2 + 1}}{2}$$

21.2562941482

Hasilnya sama dengan perhitungan menggunakan fungsi EMT.

Berikut adalah contoh lain penggunaan fungsi Maxima tersebut.

>plot2d("3*x²⁻¹,"3*x³⁻¹",xmin=-1/sqrt(3),xmax=1/sqrt(3),square=1):



images/EMT4Kalkulus_Shintia Nurcholisa

Gambar 12.283 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-223.png

>sol &= radcan(ds(3*x²⁻¹,3*x³⁻¹)); \$sol

$$3x\sqrt{9x^2 + 4}$$

>\$showev('integrate(sol,x,0,1/sqrt(3))), \$2*float(%) // panjang kurva di atas

$$3 \int_0^{\frac{1}{\sqrt{3}}} x \sqrt{9x^2 + 4} dx = 3 \left(\frac{7^{\frac{3}{2}}}{27} - \frac{8}{27} \right)$$

$$6.0 \int_{0.0}^{0.5773502691896258} x \sqrt{9.0x^2 + 4.0} dx = 2.337835372767141$$

Sikloid

Berikut kita akan menghitung panjang kurva lintasan (sikloid) suatu titik pada lingkaran yang berputar ke kanan pada permukaan datar. Misalkan jari-jari lingkaran tersebut adalah r . Posisi titik pusat lingkaran pada saat t adalah:

$$(rt, r).$$

Misalkan posisi titik pada lingkaran tersebut mula-mula $(0,0)$ dan posisinya pada saat t adalah:

$$(r(t - \sin(t)), r(1 - \cos(t))).$$

Berikut kita plot lintasan tersebut dan beberapa posisi lingkaran ketika $t=0, t=\pi/2, t=r*\pi$.

$$x = r(t - \sin(t))$$

```
[0, 1.66665833335744e-7 r, 1.33330666692022e-6 r,
4.499797504338432e-6 r, 1.066581336583994e-5 r,
2.083072932167196e-5 r, 3.599352055540239e-5 r,
5.71526624672386e-5 r, 8.530603082730626e-5 r,
1.214508019889565e-4 r, 1.665833531718508e-4 r,
2.216991628251896e-4 r, 2.877927110806339e-4 r,
3.658573803051457e-4 r, 4.568853557635201e-4 r,
5.618675264007778e-4 r, 6.817933857540259e-4 r,
8.176509330039827e-4 r, 9.704265741758145e-4 r,
0.001141105023499428 r, 0.001330669204938795 r,
0.001540100153900437 r, 0.001770376919130678 r,
0.002022476464811601 r, 0.002297373572865413 r,
0.002596040745477063 r, 0.002919448107844891 r,
0.003268563311168871 r, 0.003644351435886262 r,
0.004047774895164447 r, 0.004479793338660443 r, 0.00494136
0.005433439383882244 r, 0.005956971605131645 r,
0.006512907859185624 r, 0.007102192544548636 r,
0.007725766724910044 r, 0.00838456803503801 r,
0.009079530587017326 r, 0.009811584876838586 r, 0.01058165
0.01139067201557714 r, 0.01223954694042984 r, 0.0131291975
0.01406053493400045 r, 0.01503446588876983 r, 0.0160518930
0.01711371462093175 r, 0.01822082445851714 r, 0.0193741118
```

0.02057446139579705 r, 0.02182275311709253 r, 0.02311
 0.02446665879515308 r, 0.02586400834688696 r, 0.02731
 0.02881380207911666 r, 0.03036795126603076 r, 0.03197
 0.0336389770872163 r, 0.03535752660496472 r, 0.037132
 0.03896483946269502 r, 0.0408552420577305 r, 0.042804
 0.04481359426396048 r, 0.04688314802656623 r, 0.04901
 0.05120697598153157 r, 0.05346281777803219 r, 0.05578
 0.05816622897846346 r, 0.06061532802852698 r, 0.06313
 0.06571208837185505 r, 0.06836123997666599 r, 0.07107
 0.07386476137264342 r, 0.07672058079958999 r, 0.07964
 0.08264390910047736 r, 0.0857128256298576 r, 0.088854
 0.09206862889003742 r, 0.09535688002914089 r, 0.09871
 0.1021574371047232 r, 0.1056710629744951 r, 0.1092611
 0.1129282524731764 r, 0.1166730903725168 r, 0.1204962
 0.1243983799636342 r, 0.1283800591162231 r, 0.1324418
 0.1365844952106265 r, 0.140808431699002 r, 0.14511428
 0.1495026295080298 r, 0.1539740213994798 r]

>y &= r*(1-cos(t))

[0, 4.999958333473664e-5 r, 1.999933334222437
 4.499662510124569e-4 r, 7.998933390220841e-4 r,
 0.001249739605033717 r, 0.00179946006479581 r,
 0.002448999746720415 r, 0.003198293697380561 r,
 0.004047266988005727 r, 0.004995834721974179 r,
 0.006043902043303184 r, 0.00719136414613375 r, 0.0084
 0.009784003787362772 r, 0.01122892206395776 r, 0.0127
 0.01441523309043924 r, 0.01615630721187855 r, 0.01799
 0.01993342215875837 r, 0.02196908527585173 r, 0.02410
 0.02633360499462523 r, 0.02866202514797045 r, 0.03108
 0.03361002186548678 r, 0.03622910363410947 r, 0.03894
 0.04175612448730281 r, 0.04466351087439402 r, 0.04766
 0.05076458191755917 r, 0.0539576564716131 r, 0.057245
 0.06062728715262111 r, 0.06410317632206519 r, 0.06767
 0.07133536442348987 r, 0.07509094014268702 r, 0.07893
 0.08287917718339499 r, 0.08691105968769186 r, 0.09103
 0.09524833678003664 r, 0.09955289764732322 r, 0.10394
 0.1084317118046711 r, 0.113005077220716 r, 0.11766714

```

0.1224174381096274 r, 0.1272554923542488 r, 0.132180820322
0.1371929294852391 r, 0.1422913186361759 r, 0.147475477940
0.152744888986584 r, 0.1580990248377314 r, 0.1635373500848
0.1690593208998367 r, 0.1746643850903219 r, 0.180351982154
0.1861215433374662 r, 0.1919724916878484 r, 0.197904242115
0.2039162014509444 r, 0.2100077685026351 r, 0.216178334119
0.2224272812490723 r, 0.2287539850028937 r, 0.235157812715
0.2416381240094921 r, 0.2481942708591053 r, 0.254825597655
0.2615314412704124 r, 0.2683111311261794 r, 0.275163989259
0.2820893303890569 r, 0.2890864619877229 r, 0.296154684347
0.3032932906528349 r, 0.3105015670482534 r, 0.317778792712
0.3251242399287333 r, 0.3325371741586922 r, 0.340016854115
0.3475625318359485 r, 0.3551734527599992 r, 0.362848855801
0.3705879734263036 r, 0.3783900317293359 r, 0.386254250511
0.3941798433565377 r, 0.4021660177127022 r, 0.410211974968
0.418316910536117 r, 0.4264800139275439 r, 0.4347004688396
0.4429774532337832 r, 0.451310139418413 r]

```

Berikut kita gambar sikloid untuk $r=1$.

```

>ex &= x-sin(x); ey &= 1-cos(x); aspect(1);
>plot2d(ex,ey,xmin=0,xmax=4pi,square=1); ...
>plot2d("2+cos(x)","1+sin(x)",xmin=0,xmax=2pi,>add,color=blue);
...
>plot2d([2,ex(2)],[1,ey(2)],color=red,>add); ...
>plot2d(ex(2),ey(2),>points,>add,color=red); ...
>plot2d("2pi+cos(x)","1+sin(x)",xmin=0,xmax=2pi,>add,color=blue);
...
>plot2d([2pi,ex(2pi)],[1,ey(2pi)],color=red,>add); ...
>plot2d(ex(2pi),ey(2pi),>points,>add,color=red):

```

```
Error : [0,1.66665833335744e-7*r-sin(1.66665833335744e-7*r]
```

Error generated by error() command

adaptiveeval:

```

error(f$|" does not produce a real or column vector");
Try "trace errors" to inspect local variables after errors
plot2d:

```

```
dw/n,dw/n^2,dw/n;args());
```

Berikut dihitung panjang lintasan untuk 1 putaran penuh.
(Jangan salah menduga bahwa panjang lintasan 1 putaran penuh sama dengan keliling lingkaran!)

```
>ds &= radcan(sqrt(diff(ex,x)^2+diff(ey,x)^2)); $ds=trigsimp(ds) //
elemen panjang kurva sikloid
```

Maxima said:

```
diff: second argument must be a variable; found errexp1
-- an error. To debug this try: debugmode(true);
```

Error in:

```
ds &= radcan(sqrt(diff(ex,x)^2+diff(ey,x)^2)); $ds
^
```

```
>ds &= trigsimp(ds); $ds
```

```
>$showev('integrate(ds,x,0,2*pi)) // hitung panjang sikloid satu
putaran penuh
```

Maxima said:

```
defint: variable of integration must be a simple or s
defint: found errexp1
#0: showev(f='integrate(ds,[0,1.66665833335744e-7*r,1
-- an error. To debug this try: debugmode(true);
```

Error in:

```
$showev('integrate(ds,x,0,2*pi)) // hitung panjang si
^
```

```
>integrate(mxm("ds"),0,2*pi) // hitung secara numerik
```

Illegal function result in map.

```
%evalexpression:
```

```
if maps then return %mapexpression1(x,f$;args());
```

```
gauss:
```

```
if maps then y=%evalexpression(f$,a+h-(h*xn)',map
```

```

adaptivegauss:
    t1=gauss(f$,c,c+h;args(),=maps);
Try "trace errors" to inspect local variables after errors
integrate:
    return adaptivegauss(f$,a,b,eps*1000;args(),=maps);

>romberg(mxm("ds"),0,2*pi) // cara lain hitung secara numerik

```

Wrong argument!

Cannot combine a symbolic expression here.
Did you want to create a symbolic expression?
Then start with `&`.

```

Try "trace errors" to inspect local variables after errors
romberg:
    if cols(y)==1 then return y*(b-a); endif;
Error in:
romberg(mxm("ds"),0,2*pi) // cara lain hitung secara numerik

```

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 2π 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 &= r*cos(t); fy &=r*sin(t);
  >&assume(t>0,r>0); s &=integrate(sqrt(diff(fx,t)^2+diff(fy,t)^2),t,0,t);
s // elemen panjang kurva, panjang busur lingkaran (s)
```

Maxima said:

```
diff: second argument must be a variable; found errexp1
-- an error. To debug this try: debugmode(true);
```

Error in:

```
... =integrate(sqrt(diff(fx,t)^2+diff(fy,t)^2),t,0,t); s /
```

```
>&kill(s); fx &= r*cos(s/r); fy &=r*sin(s/r); // definisi ulang
persamaan parametrik terhadap s dengan substitusi t=s/r
```

```
>k &= trigsimp(sqrt(diff(fx,s,2)^2+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

$$\begin{aligned}\kappa &= \frac{d\phi}{ds} = \frac{\frac{d\phi}{dt}}{\frac{ds}{dt}} \quad (\phi \text{ adalah sudut kemiringan garis singgung dan } s \text{ adalah panjang busur}) \\ &= \frac{\frac{d\phi}{dt}}{\sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2}} = \frac{\frac{d\phi}{dt}}{\sqrt{x'(t)^2 + y'(t)^2}}.\end{aligned}$$

Selanjutnya, pembilang pada persamaan di atas dapat dicari sebagai berikut.

$$\sec^2 \phi \frac{d\phi}{dt} = \frac{d}{dt} (\tan \phi) = \frac{d}{dt} \left(\frac{dy}{dx} \right) = \frac{d}{dt} \left(\frac{dy/dt}{dx/dt} \right) = \frac{d}{dt} \left(\frac{y'(t)}{x'(t)} \right) = \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2}$$

$$\begin{aligned}\frac{d\phi}{dt} &= \frac{1}{\sec^2 \phi} \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2} \\ &= \frac{1}{1 + \tan^2 \phi} \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2} \\ &= \frac{1}{1 + \left(\frac{y'(t)}{x'(t)}\right)^2} \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2} \\ &= \frac{x'(t)y''(t) - x''(t)y'(t)}{x'(t)^2 + y'(t)^2}.\end{aligned}$$

Jadi, rumus kurvatur untuk kurva parametrik

$$x = x(t), \quad y = y(t)$$

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

$$x = t, \quad y = f(t), \quad \text{dengan } x'(t) = 1, \quad x''(t) = 0,$$

sehingga kurvturnya adalah

$$\kappa(t) = \frac{y''(t)}{(1 + y'(t)^2)^{3/2}}.$$

Contoh:

Akan ditentukan kurvatur parabola

$$y = ax^2 + bx + c.$$

>function f(x) &= a*x^2+b*x+c; \$y=f(x)

```
>function k(x) &= (diff(f(x),x,2))/(1+diff(f(x),x)^2)^(3/2); $'k(x)=k(x) //  
kelengkungan parabola
```

Maxima said:

```
diff: second argument must be a variable; found errexp  
-- an error. To debug this try: debugmode(true);
```

Error in:

```
... (x) &= (diff(f(x),x,2))/(1+diff(f(x),x)^2)^(3/2)
```

```
>function f(x) &= x^2+x+1; $y=f(x) // akan kita plot kelengkungan  
parabola untuk a=b=c=1
```

Error : f(x) does not produce a real or column vector

Error generated by error() command

%ploteval:

error(f\$|" does not produce a real or column vector");

adaptiveevalone:

s=%ploteval(g\$,t;args());

Try "trace errors" to inspect local variables after errors

plot2d:

dw/n,dw/n^2,dw/n,auto;args());

Untuk kurva yang dinyatakan dengan fungsi implisit

$$F(x, y) = 0$$

dengan turunan-turunan parsial

$$F_x = \frac{\partial F}{\partial x}, F_y = \frac{\partial F}{\partial y}, F_{xy} = \frac{\partial}{\partial y} \left(\frac{\partial F}{\partial x} \right), F_{xx} = \frac{\partial}{\partial x} \left(\frac{\partial F}{\partial x} \right), F_{yy} = \frac{\partial}{\partial y} \left(\frac{\partial F}{\partial y} \right),$$

berlaku

$$F_x dx + F_y dy = 0 \text{ atau } \frac{dy}{dx} = -\frac{F_x}{F_y},$$

sehingga kurvturnya adalah

$$\kappa = \frac{F_y^2 F_{xx} - 2F_x F_y F_{xy} + F_x^2 F_{yy}}{(F_x^2 + F_y^2)^{3/2}}.$$

(Silakan Anda turunkan sendiri!)

Contoh 1:

Parabola

$$y = ax^2 + bx + c$$

dapat dinyatakan ke dalam persamaan implisit

$$ax^2 + bx + c - y = 0.$$

```
>function F(x,y) &=a*x^2+b*x+c-y; $F(x,y)
```

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).

- Gambarkan kurva asli, kurva kurvatur, kurva jari-jari lingkaran oskulasi, dan salah satu lingkaran oskulasinya.
1. Tentukan panjang kurva dan volume benda putar kurva $y=f(x)$ yang diputar mengelilingi sumbu x dari $x=0$ sampai $x=4$

$$f(x) = x^3$$

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

[0,4.629560185733296 × 10⁻²¹ r³, 2.370228152344803 × 10⁻¹⁸ r³, 9.11126989421

>\$showev('integrate(pi*(f(x))^2,x,0,4)) // volume benda putar

Maxima said:

defint: variable of integration must be a simple or subscr

BARISAN DAN DERET

(Catatan: bagian ini belum lengkap. Anda dapat membaca contoh-contoh penggunaan EMT dan Maxima untuk menghitung limit barisan, rumus jumlah parsial suatu deret, jumlah tak hingga suatu deret konvergen, dan sebagainya. Anda dapat mengeksplor contoh-contoh di EMT atau berbagai panduan penggunaan Maxima di software Maxima atau dari Internet.)

Barisan dapat didefinisikan dengan beberapa cara di dalam EMT, di antaranya: dengan cara yang sama seperti mendefinisikan vektor dengan elemen-elemen beraturan (menggunakan titik dua “:”); menggunakan perintah “sequence” dan rumus barisan (suku ke -n); menggunakan perintah “iterate” atau “niterate”; menggunakan fungsi Maxima “create_list” atau “makelist” untuk menghasilkan barisan simbolik; menggunakan fungsi biasa yang inputnya vektor atau barisan; menggunakan fungsi rekursif.

EMT menyediakan beberapa perintah (fungsi) terkait barisan, yakni:

- sum: menghitung jumlah semua elemen suatu barisan
- cumsum: jumlah kumulatif suatu barisan
- differences: selisih antar elemen-elemen berturutan

EMT juga dapat digunakan untuk menghitung jumlah deret berhingga maupun deret tak hingga, dengan menggunakan perintah (fungsi) “sum”. Perhitungan dapat dilakukan secara numerik maupun simbolik dan eksak.

Berikut adalah beberapa contoh perhitungan barisan dan deret menggunakan EMT.

```
>1:10 // barisan sederhana
```

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

```
>1:2:30
```

[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25]

ITERASI DAN BARISAN

EMT menyediakan fungsi `iterate("g(x)", x0, n)` untuk melakukan iterasi

$$x_{k+1} = g(x_k), \quad x_0 = x_0, \quad k = 1, 2, 3, \dots, n.$$

Berikut ini disajikan contoh-contoh penggunaan iterasi dan rekursi dengan EMT. Contoh pertama menunjukkan pertumbuhan dari nilai awal 1000 dengan laju pertambahan 5%, selama 10 periode.

```
>q=1.05; iterate("x*q",1000,n=10)
```

```
1000
1050
1102.5
1157.63
1215.51
1276.28
1340.1
1407.1
1477.46
1551.33
1628.89
```

Contoh berikutnya memperlihatkan bahaya menabung di bank pada masa sekarang! Dengan bunga tabungan sebesar 6% per tahun atau 0.5% per bulan dipotong pajak 20%, dan biaya administrasi 10000 per bulan, tabungan sebesar 1 juta tanpa diambil selama sekitar 10 tahunan akan habis diambil oleh bank!

```
>r=0.005; plot2d(iterate("(1+0.8*r)*x-10000",1000000,n=130)):
```

Silakan Anda coba-coba, dengan tabungan minimal berapa agar tidak akan habis diambil oleh bank dengan ketentuan bunga dan biaya administrasi seperti di atas.

Berikut adalah perhitungan minimal tabungan agar aman di bank dengan bunga sebesar r dan biaya administrasi a , pajak bunga 20%.



Gambar 12.285 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-271.png

```
>$solve(0.8*r*A-a,A), $% with [r=0.005, a=10]
```

$$\left[A = \frac{5a}{4r} \right]$$

$$[A = 2500.0]$$

Berikut didefinisikan fungsi untuk menghitung saldo tabungan, kemudian dilakukan iterasi.

```
>function saldo(x,r,a) := round((1+0.8*r)*x-a,2);
>iterate({{"saldo",0.005,10}},1000,n=6)
```

```
[1000, 994, 987.98, 981.93, 975.86, 969.76, 963.64]
```

```
>iterate({{"saldo",0.005,10}},2000,n=6)
```

```
[2000, 1998, 1995.99, 1993.97, 1991.95, 1989.92, 1987.90]
```

```
>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, 3012.10]
```

Tabungan yang lebih dari 2,5 juta baru akan bertambah jika tidak ada penarikan.

Untuk barisan yang lebih kompleks dapat digunakan fungsi “sequence()”. Fungsi ini menghitung nilai-nilai $x[n]$ dari semua nilai sebelumnya, $x[1], \dots, x[n-1]$ yang diketahui.

Berikut adalah contoh barisan Fibonacci.

$$x_n = x_{n-1} + x_{n-2}, \quad x_1 = 1, \quad x_2 = 1$$

```
>sequence(“x[n-1]+x[n-2]”,[1,1],15)
```

```
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144]
```

Barisan Fibonacci memiliki banyak sifat menarik, salah satunya adalah akar pangkat ke- n suku ke- n akan konvergen ke pecahan emas:

$$>\$(1+\text{sqrt}(5))/2=\text{float}((1+\text{sqrt}(5))/2)$$

$$\frac{\sqrt{5} + 1}{2} = 1.618033988749895$$

```
>plot2d(sequence(“x[n-1]+x[n-2]”,[1,1],250)^(1/(1:250))):
```



images/EMT4Kalkulus_Shintia Nurchol

Gambar 12.286 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-276.png

Barisan yang sama juga dapat dihasilkan dengan menggunakan loop.

```
>x=ones(500); for k=3 to 500; x[k]=x[k-1]+x[k-2]; end;
```

Rekursi dapat dilakukan dengan menggunakan rumus yang tergantung pada semua elemen sebelumnya. Pada contoh berikut, elemen ke- n merupakan jumlah $(n-1)$ elemen sebelumnya, dimulai dengan 1 (elemen ke-1). Jelas, nilai elemen ke- n adalah $2^{(n-2)}$, 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_of_Theodorus.png

Spiral Theodorus (spiral segitiga siku-siku) dapat digambar secara rekursif. Rumus rekursifnya adalah:

$$x_n = \left(1 + \frac{i}{\sqrt{n-1}}\right) x_{n-1}, \quad x_1 = 1,$$

yang menghasilkan barisan bilangan kompleks.

```
>function g(n) := 1+I/sqrt(n)
```

Rekursinya dapat dijalankan sebanyak 17 untuk menghasilkan barisan 17 bilangan kompleks, kemudian digambar bilangan-bilangan kompleksnya.

```
>x=sequence("g(n-1)*x[n-1]",1,17); plot2d(x,r=3.5); textbox(latex("Spiral Theodorus"),0.4):
```



Gambar 12.287 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-280.png

Selanjutnya dihubungkan titik 0 dengan titik-titik kompleks tersebut menggunakan loop.

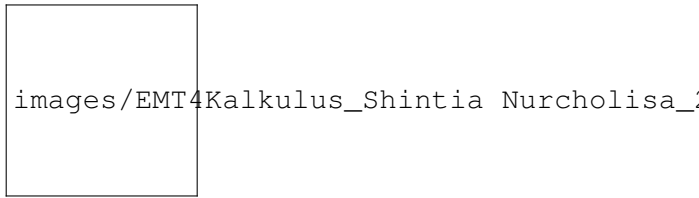
```
>for i=1:cols(x); plot2d([0,x[i]],>add); end;
>
```

Spiral tersebut juga dapat didefinisikan menggunakan fungsi rekursif, yang tidak memerlukan indeks dan bilangan kompleks. Dalam hal ini digunakan vektor kolom pada bidang.

```
>function gstep (v) ...
```

```
w=[-v[2]; v[1]];

```

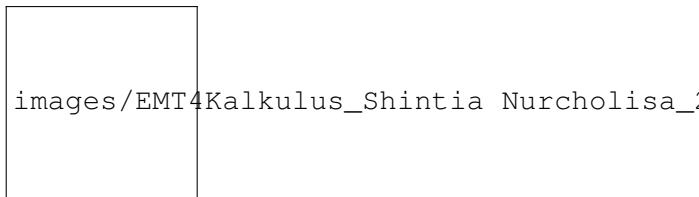


Gambar 12.288 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-281.png

```
return v+w/norm(w);
endfunction
```

Jika dilakukan iterasi 16 kali dimulai dari $[1;0]$ akan didapatkan matriks yang memuat vektor-vektor dari setiap iterasi.

```
>x=iterate("gstep",[1;0],16); plot2d(x[1],x[2],r=3.5,>points):
```



Gambar 12.289 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-282.png

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 meingkatkan perhitungan pada ekspresi dapat digunakan pembagian intervalnya, menggunakan fungsi `ieval()`.

```
>function s(x) := ieval("(x+2/x)/2",x,10)
```

Selanjutnya dapat dilakukan iterasi hingga diperoleh hasil optimal, dan intervalnya tidak semakin mengecil. Hasilnya berupa interval yang memuat akar persamaan:

$$x = \frac{1}{2} \left(x + \frac{2}{x} \right).$$

Satu-satunya solusi adalah

$$x = \sqrt{2}.$$

```
>iterate("s",1,2)
```

```
~1.41421356236,1.41421356239~
```

Fungsi `"iterate()"` juga dapat bekerja pada vektor. Berikut adalah contoh fungsi vektor, yang menghasilkan rata-rata aritmetika dan rata-rata geometri.

$$(a_{n+1}, b_{n+1}) = \left(\frac{a_n + b_n}{2}, \sqrt{a_n b_n} \right)$$

Iterasi ke-n disimpan pada vektor kolom x[n].

```
>function g(x) := [(x[1]+x[2])/2;sqrt(x[1]*x[2])]
```

Iterasi dengan menggunakan fungsi tersebut akan konvergen ke rata-rata aritmetika dan geometri dari nilai-nilai awal.

```
>iterate("g",[1;5])
```

```
2.60401
```

```
2.60401
```

Hasil tersebut konvergen agak cepat, seperti kita cek sebagai berikut.

```
>iterate("g",[1;5],4)
```

1	3	2.61803	2.60401
5	2.23607	2.59002	2.60301

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
```

```
~0.999999999999999778,1.000000000000000022~ ...
~4.99999999999999911,5.000000000000000089~ ...
```

Iterasi berikut konvergen sangat lambat.

```
>iterate("sqrt(x)",2,10)
```

```
[2, 1.41421, 1.18921, 1.09051, 1.04427, 1.0219,
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/EMT4Kalkulus_Shintia Nurchol

Gambar 12.290 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-287.png

```
>A =[0.5,0.2;0.7,0.1]; b=[2;2]; ...
```

```
> x=[1;1]; repeat xnew=A.x-b; until all(xnew~x); x=xnew; end; ...
```

```
> x,
```

```
-7.09677
```

```
-7.74194
```


ITERASI DI DALAM FUNGSI

Fungsi atau program juga dapat menggunakan iterasi dan dapat digunakan untuk melakukan iterasi. Berikut adalah beberapa contoh iterasi di dalam fungsi.

Contoh berikut adalah suatu fungsi untuk menghitung berapa lama suatu iterasi konvergen. Nilai fungsi tersebut adalah hasil akhir iterasi dan banyak iterasi sampai konvergen.

```
>function map hiter(f$,x0) ...
```

```
x=x0;
maxiter=0;
repeat
  xnew=f$(x);
  maxiter=maxiter+1;
until xnew~x;
x=xnew;
end;
return maxiter;
endfunction
```

Misalnya, berikut adalah iterasi untuk mendapatkan hampiran akar kuadrat 2, cukup cepat, konvergen pada iterasi ke-5, jika dimulai dari hampiran awal 2.

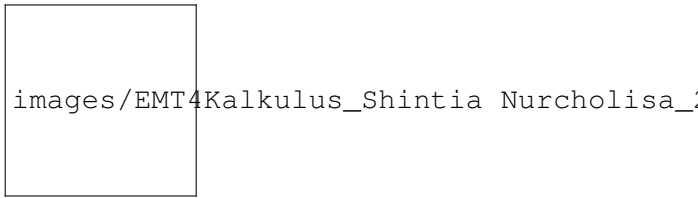
```
>hiter("(x+2/x)/2",2)
```

5

Karena fungsinya didefinisikan menggunakan "map". maka nilai awalnya dapat berupa vektor.

```
>x=1.5:0.1:10; hasil=hiter("(x+2/x)/2",x); ...
> plot2d(x,hasil):
```

Dari gambar di atas terlihat bahwa kekonvergenan iterasinya semakin lambat, untuk nilai awal semakin besar, namun penambahannya



Gambar 12.291 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-288.png

tidak kontinu. Kita dapat menemukan kapan maksimum iterasinya bertambah.

```
>hasil[1:10]
```

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

```
>x[nonzeros(differences(hasil))]
```

```
[1.5, 2, 3.4, 6.6]
```

maksimum iterasi sampai konvergen meningkat pada saat nilai awalnya 1.5, 2, 3.4, dan 6.6.

Contoh berikutnya adalah metode Newton pada polinomial kompleks berderajat 3.

```
>p &= x^3-1; newton &= x-p/diff(p,x); $newton
```

Maxima said:

```
diff: second argument must be a variable; found errexp1
-- an error. To debug this try: debugmode(true);
```

Error in:

```
p &= x^3-1; newton &= x-p/diff(p,x); $newton ...
```

Selanjutnya didefinisikan fungsi untuk melakukan iterasi (aslinya 10 kali).

```
>function iterasi(f$,x,n=10) ...
```

```

loop 1 to n; x=f$(x); end;
return x;
endfunction

```

Kita mulai dengan menentukan titik-titik grid pada bidang kompleksnya.

```
>r=1.5; x=linspace(-r,r,501); Z=x+I*x'; W=iterasi(newton,Z);
```

Function newton needs at least 3 arguments!

Use: newton (f\$: call, df\$: call, x: scalar complex {

Error in:

```
... x=linspace(-r,r,501); Z=x+I*x'; W=iterasi(newton,
```

Berikut adalah akar-akar polinomial di atas.

```
>z=&solve(p)()
```

Maxima said:

solve: more equations than unknowns.

Unknowns given :

```
[r]
```

Equations given:

```
errexpl
```

```
-- an error. To debug this try: debugmode(true);
```

Error in:

```
z=&solve(p)() ...
^
```

Untuk menggambar hasil iterasinya, dihitung jarak dari hasil iterasi ke-10 ke masing-masing akar, kemudian digunakan untuk menghitung warna yang akan digambar, yang menunjukkan limit untuk masing-masing nilai awal.

Fungsi plotrgb() menggunakan jendela gambar terkini untuk menggambar warna RGB sebagai matriks.

```
>C=rgb(max(abs(W-z[1]),1),max(abs(W-z[2]),1),max(abs(W-
z[3]),1)); ...
```

```
> plot2d(none,-r,r,-r,r); plotrgb(C):
```

Variable W not found!

Error in:

```
C=rgb(max(abs(W-z[1]),1),max(abs(W-z[2]),1),max(abs(W-z[3]),1))
```

ITERASI SIMBOLIK

Seperti sudah dibahas sebelumnya, untuk menghasilkan barisan ekspresi simbolik dengan Maxima dapat digunakan fungsi makelist().

>powerdisp:true // untuk menampilkan deret pangkat mulai dari suku berpangkat terkecil

true

>deret &= makelist(taylor(exp(x),x,0,k),k,1,3); \$deret // barisan deret Taylor untuk e^x

Maxima said:

```
taylor: 0.1539740213994798*r cannot be a variable.  
-- an error. To debug this try: debugmode(true);
```

Error in:

```
deret &= makelist(taylor(exp(x),x,0,k),k,1,3); $deret  
^
```

Untuk mengubah barisan deret tersebut menjadi vektor string di EMT digunakan fungsi mxm2str(). Selanjutnya, vektor string/ekspresi hasilnya dapat digambar seperti menggambar vektor ekspresi pada EMT.

>plot2d("exp(x)",0,3); // plot fungsi aslinya, e^x

>plot2d(mxm2str("deret"),>add,color=4:6): // plot ketiga deret Taylor hampiran fungsi tersebut

Maxima said:

```
length: argument cannot be a symbol; found deret  
-- an error. To debug this try: debugmode(true);
```

mxmeval:

```
return evaluate(mxm(s));
```

Try "trace errors" to inspect local variables after e

```
mxm2str:
    n=mxmeval("length(VVV)");
```

Selain cara di atas dapat juga dengan cara menggunakan indeks pada vektor/list yang dihasilkan.

```
>$deret[3]
```

deret₃

```
>plot2d(["exp(x)",&deret[1],&deret[2],&deret[3]],0,3,color=1:4):
```

```
deret is not a variable!
```

```
Error in expression: deret[1]
```

```
%ploteval:
```

```
    y0=f$(x[1],args());
```

```
Try "trace errors" to inspect local variables after errors
```

```
plot2d:
```

```
    u=u_(%ploteval(xx[#],t;args()));
```

```
>$sum(sin(k*x)/k,k,1,5)
```

$$\sin(x) + \frac{\sin(3x)}{3} + \frac{\sin(5x)}{5} + \frac{\sin(7x)}{7} + \frac{\sin(9x)}{9} + \frac{\sin(11x)}{11} + \dots$$

Berikut adalah cara menggambar kurva

$$y = \sin(x) + \frac{\sin 3x}{3} + \frac{\sin 5x}{5} + \dots$$

```
>plot2d(&sum(sin((2*k+1)*x)/(2*k+1),k,0,20),0,2pi):
```

Hal serupa juga dapat dilakukan dengan menggunakan matriks, misalkan kita akan menggambar kurva

$$y = \sum_{k=1}^{100} \frac{\sin(kx)}{k}, \quad 0 \leq x \leq 2\pi.$$

```
>x=linspace(0,2pi,1000); k=1:100; y=sum(sin(k*x')/k)'; plot2d(x,y):
```



Gambar 12.292 images/EMT4Kalkulus_Shintia%20Nurcholisa_23030630071-293.png

TABEL FUNGSI

Terdapat cara menarik untuk menghasilkan barisan dengan ekspresi Maxima. Perintah `mxmtable()` berguna untuk menampilkan dan menggambar barisan dan menghasilkan barisan sebagai vektor kolom.

Sebagai contoh berikut adalah barisan turunan ke- n x^x di $x=1$.

```
>mxmtable("diffat(x^x,x=1,n)","n",1,8,frac=1);
```

Maxima said:

```
diff: second argument must be a variable; found errexpr
#0: diffat(expr=[0,1.66665833335744e-7*r,1.333306666692
-- an error. To debug this try: debugmode(true);
```

```
%mxmevtable:
```

```
return mxm("@expr,@var=@value")();
```

Try "trace errors" to inspect local variables after error.

```
mxmtable:
```

```
y[#,1]=%mxmevtable(expr,var,x[#]);
```

```
>$'sum(k, k, 1, n) = factor(ev(sum(k, k, 1, n),simpsum=true)) //
```

`simpsum`:menghitung deret secara simbolik

$$\sum_{k=1}^n k = \frac{n(1+n)}{2}$$

```
>$'sum(1/(3^k+k), k, 0, inf) = factor(ev(sum(1/(3^k+k), k, 0,
inf),simpsum=true))
```

$$\sum_{k=0}^{\infty} \frac{1}{k+3^k} = \sum_{k=0}^{\infty} \frac{1}{k+3^k}$$

Di sini masih gagal, hasilnya tidak dihitung.

```
>$'sum(1/x^2, x, 1, inf)= ev(sum(1/x^2, x, 1, inf),simpsum=true)
```

// ev: menghitung nilai ekspresi

$$\sum_{x=1}^{\infty} \frac{1}{x^2} = \frac{\pi^2}{6}$$

>\$'sum((-1)^(k-1)/k, k, 1, inf) = factor(ev(sum((-1)^(x-1)/x, x, 1, inf),simpsum=true))

$$\sum_{k=1}^{\infty} \frac{(-1)^{-1+k}}{k} = - \sum_{x=1}^{\infty} \frac{(-1)^x}{x}$$

Di sini masih gagal, hasilnya tidak dihitung.

>\$'sum((-1)^k/(2*k-1), k, 1, inf) = factor(ev(sum((-1)^k/(2*k-1), k, 1, inf),simpsum=true))

$$\sum_{k=1}^{\infty} \frac{(-1)^k}{-1+2k} = \sum_{k=1}^{\infty} \frac{(-1)^k}{-1+2k}$$

>\$ev(sum(1/n!, n, 0, inf),simpsum=true)

$$\sum_{n=0}^{\infty} \frac{1}{n!}$$

Di sini masih gagal, hasilnya tidak dihitung, harusnya hasilnya e.

>&assume(abs(x)<1); \$'sum(a*x^k, k, 0, inf)=ev(sum(a*x^k, k, 0, inf),simpsum=true), &forget(abs(x)<1);

Answering "Is -94914474571+15819*r positive, negative or zero?"

Maxima said:

sum: sum is divergent.

-- an error. To debug this try: debugmode(true);

Error in:

... k, 0, inf)=ev(sum(a*x^k, k, 0, inf),simpsum=true), &assume(abs(x)<1);

Deret geometri tak hingga, dengan asumsi rasional antara -1 dan

1.

>\$'sum(x^k/k!,k,0,inf)=ev(sum(x^k/k!,k,0,inf),simpsum=true)

$$\left[0, \sum_{k=0}^{\infty} \frac{(1.66665833335744 \times 10^{-7})^k r^k}{k!}, \sum_{k=0}^{\infty} \frac{(1.33330666692022 \times 10^{-6})^k}{k!} \right]$$

>\$limit(sum(x^k/k!,k,0,n),n,inf)

$$\left[0, \lim_{n \rightarrow \infty} \sum_{k=0}^n \frac{(1.66665833335744 \times 10^{-7})^k r^k}{k!}, \lim_{n \rightarrow \infty} \sum_{k=0}^n \frac{(1.33330666692022 \times 10^{-6})^k}{k!} \right]$$

>function d(n) &= sum(1/(k^2-k),k,2,n); \$'d(n)=d(n)

$$d(n) = \sum_{k=2}^n \frac{1}{-k + k^2}$$

>\$d(10)=ev(d(10),simpsum=true)

$$\sum_{k=2}^{10} \frac{1}{-k + k^2} = \frac{9}{10}$$

>\$d(100)=ev(d(100),simpsum=true)

$$\sum_{k=2}^{100} \frac{1}{-k + k^2} = \frac{99}{100}$$

Deret Taylor

Deret Taylor suatu fungsi f yang diferensiabel sampai tak hingga di sekitar x=a adalah:

$$f(x) = \sum_{k=0}^{\infty} \frac{(x-a)^k f^{(k)}(a)}{k!}.$$

>\$'e^x=taylor(exp(x),x,0,10)// deret Taylor e^x di sekitar x=0, sampai suku ke-11

Maxima said:

taylor: 0.1539740213994798*r cannot be a variable.

```
-- an error. To debug this try: debugmode(true);
```

Error in:

```
$'e^x =taylor(exp(x),x,0,10) // deret Taylor e^x di sekitar x=0
```

```
>$'log(x)=taylor(log(x),x,1,10)// deret log(x) di sekitar x=1
```

Maxima said:

```
log: encountered log(0).
```

```
-- an error. To debug this try: debugmode(true);
```

Error in:

```
$'log(x)=taylor(log(x),x,1,10)// deret log(x) di sekitar x=1
```

unicodehyperref hyphensurl []book xcolor amsmath,amssymb
iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype
[protrusion]basicmath parskip graphicx bookmark xurl

VISUALISASI DAN PERHITUNGAN GEOMETRI DENGAN EMT

Euler menyediakan beberapa fungsi untuk melakukan visualisasi dan perhitungan geometri, baik secara numerik maupun analitik (seperti biasanya tentunya, menggunakan Maxima). Fungsi-fungsi untuk visualisasi dan perhitungan geometri tersebut disimpan di dalam file program “geometry.e”, sehingga file tersebut harus dipanggil sebelum menggunakan fungsi-fungsi atau perintah-perintah untuk geometri.

```
>load geometry
```

```
Numerical and symbolic geometry.
```

Fungsi-fungsi Geometri

Fungsi-fungsi untuk Menggambar Objek Geometri:

`defaultd:=textheight()*1.5`: nilai asli untuk parameter d
`setPlotrange(x1,x2,y1,y2)`: menentukan rentang x dan y pada bidang koordinat
`setPlotRange(r)`: pusat bidang koordinat (0,0) dan batas-batas sumbu-x dan y adalah -r sd r
`plotPoint (P, “P”)`: menggambar titik P dan diberi label “P”
`plotSegment (A,B, “AB”, d)`: menggambar ruas garis AB, diberi label “AB” sejauh d
`plotLine (g, “g”, d)`: menggambar garis g diberi label “g” sejauh d
`plotCircle (c, “c”,v,d)`: Menggambar lingkaran c dan diberi label “c”
`plotLabel (label, P, V, d)`: menuliskan label pada posisi P

Fungsi-fungsi Geometri Analitik (numerik maupun simbolik):

`turn(v, phi)`: memutar vektor v sejauh phi
`turnLeft(v)`: memutar vektor v ke kiri
`turnRight(v)`: memutar vektor v ke kanan
`normalize(v)`: normal vektor v
`crossProduct(v, w)`: hasil kali silang vektor v dan w.
`lineThrough(A, B)`: garis melalui A dan B, hasilnya [a,b,c] sdh. $ax+by=c$.

`lineWithDirection(A,v)`: garis melalui A searah vektor v
`getLineDirection(g)`: vektor arah (gradien) garis g
`getNormal(g)`: vektor normal (tegak lurus) garis g
`getPointOnLine(g)`: titik pada garis g
`perpendicular(A, g)`: garis melalui A tegak lurus garis g
`parallel(A, g)`: garis melalui A sejajar garis g
`lineIntersection(g, h)`: titik potong garis g dan h
`projectToLine(A, g)`: proyeksi titik A pada garis g
`distance(A, B)`: jarak titik A dan B
`distanceSquared(A, B)`: kuadrat jarak A dan B
`quadrance(A, B)`: kuadrat jarak A dan B
`areaTriangle(A, B, C)`: luas segitiga ABC
`computeAngle(A, B, C)`: besar sudut $\angle ABC$
`angleBisector(A, B, C)`: garis bagi sudut $\angle ABC$
`circleWithCenter(A, r)`: lingkaran dengan pusat A dan jari-jari r
`getCircleCenter(c)`: pusat lingkaran c
`getCircleRadius(c)`: jari-jari lingkaran c
`circleThrough(A,B,C)`: lingkaran melalui A, B, C
`middlePerpendicular(A, B)`: titik tengah AB
`lineCircleIntersections(g, c)`: titik potong garis g dan lingkaran c
`circleCircleIntersections(c1, c2)`: titik potong lingkaran $c1$ dan $c2$
`planeThrough(A, B, C)`: bidang melalui titik A, B, C

Fungsi-fungsi Khusus Untuk Geometri Simbolik:

`getLineEquation(g,x,y)`: persamaan garis g dinyatakan dalam x dan y
`getHesseForm(g,x,y,A)`: bentuk Hesse garis g dinyatakan dalam x dan y dengan titik A pada sisi positif (kanan/atas) garis
`quad(A,B)`: kuadrat jarak AB
`spread(a,b,c)`: Spread segitiga dengan panjang sisi-sisi a,b,c , yakni $\sin(\alpha)^2$ dengan α sudut yang menghadap sisi a .
`crosslaw(a,b,c,sa)`: persamaan 3 quads dan 1 spread pada segitiga dengan panjang sisi a, b, c .
`triplespread(sa,sb,sc)`: persamaan 3 spread sa,sb,sc yang membentuk suatu segitiga
`doublespread(sa)`: Spread sudut rangkap Spread 2ϕ , dengan

$sa = \sin(\phi)^2 \text{ spread } a$.

Contoh 1: Luas, Lingkaran Luar, Lingkaran Dalam Segitiga

Untuk menggambar objek-objek geometri, langkah pertama adalah menentukan rentang sumbu-sumbu koordinat. Semua objek geometri akan digambar pada satu bidang koordinat, sampai didefinisikan bidang koordinat yang baru.

```
>setPlotRange(-0.5,2.5,-0.5,2.5); // mendefinisikan bidang koordinat baru
```

Sekarang tetapkan tiga titik dan plotlah.

```
>A=[1,0]; plotPoint(A,"A"); // definisi dan gambar tiga titik
```

```
>B=[0,1]; plotPoint(B,"B");
```

```
>C=[2,2]; plotPoint(C,"C");
```

Kemudian tiga segmen.

```
>plotSegment(A,B,"c"); // c=AB
```

```
>plotSegment(B,C,"a"); // a=BC
```

```
>plotSegment(A,C,"b"); // b=AC
```

Fungsi geometri mencakup fungsi untuk membuat garis dan lingkaran. Format untuk garis adalah $[a,b,c]$, yang merepresentasikan garis dengan persamaan $ax+by=c$.

```
>lineThrough(B,C) // garis yang melalui B dan C
```

```
[-1, 2, 2]
```

Hitung garis tegak lurus yang melalui A pada BC.

```
>h=perpendicular(A,lineThrough(B,C)); // garis h tegak lurus BC melalui A
```

Dan persimpangannya dengan BC.

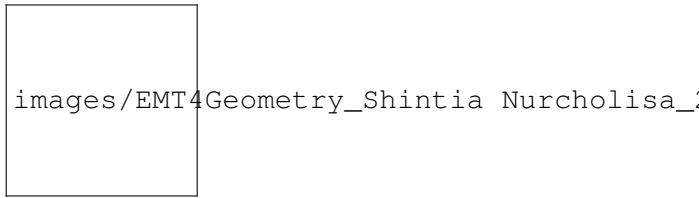
```
>D=lineIntersection(h,lineThrough(B,C)); // D adalah titik potong h dan BC
```

Rencanakan itu.

```
>plotPoint(D,value=1); // koordinat D ditampilkan
```

```
>aspect(1); plotSegment(A,D); // tampilkan semua gambar hasil plot...()
```

Hitung luas ABC:



Gambar 12.293 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-001.png

```
>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 pada C.

```
>degprint(computeAngle(B,C,A))
```

36°52'11.63''

Sekarang, lingkarilah segitiga tersebut.

```
>c=circleThrough(A,B,C); // lingkaran luar segitiga ABC
```

```
>R=getCircleRadius(c); // jari2 lingkaran luar
```

```
>O=getCircleCenter(c); // titik pusat lingkaran c
```

```
>plotPoint(O,"O"); // gambar titik "O"
```

```
>plotCircle(c,"Lingkaran luar segitiga ABC");
```

Tampilkan koordinat titik pusat dan jari-jari lingkaran luar.

```
>O, R
```



images/EMT4Geometry_Shintia Nurchol

Gambar 12.294 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-002.png

```
[1.16667, 1.16667]  
1.17851130198
```

Sekarang akan digambar lingkaran dalam segitiga ABC. Titik pusat lingkaran dalam adalah titik potong garis-garis bagi sudut.

```
>l=angleBisector(A,C,B); // garis bagi <ACB  
>g=angleBisector(C,A,B); // garis bagi <CAB  
>P=lineIntersection(l,g) // titik potong kedua garis bagi sudut
```

```
[0.86038, 0.86038]
```

Tambahkan semuanya ke plot.

```
>color(5); plotLine(l); plotLine(g); color(1); // gambar kedua  
garis bagi sudut
```

```
>plotPoint(P,"P"); // gambar titik potongnya
```

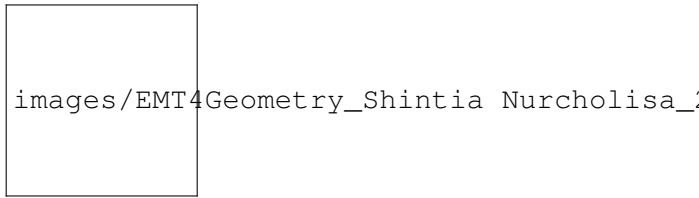
```
>r=norm(P-projectToLine(P,lineThrough(A,B))) // jari-jari  
lingkaran dalam
```

```
0.509653732104
```

```
>plotCircle(circleWithCenter(P,r),"Lingkaran dalam segitiga  
ABC"); // gambar lingkaran dalam
```

Latihan

1. Tentukan ketiga titik singgung lingkaran dalam dengan sisi-sisi segitiga ABC.

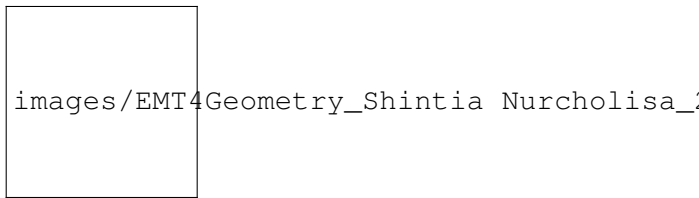


Gambar 12.295 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-003.png

```
>setPlotRange(-2.5,4.5,-2.5,4.5);
>A=[-2,1]; plotPoint(A,"A");
>B=[1,-2]; plotPoint(B,"B");
>C=[4,4]; plotPoint(C,"C");
```

2. Gambar segitiga dengan titik-titik sudut ketiga titik singgung tersebut. Merupakan segitiga apakah itu?

```
>plotSegment(A,B,"c")
>plotSegment(B,C,"a")
>plotSegment(A,C,"b")
>aspect(1):
```

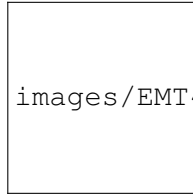


Gambar 12.296 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-004.png

```
>l=angleBisector(A,C,B);
>g=angleBisector(C,A,B);
>P=lineIntersection(l,g)
```

```
[0.581139, 0.581139]
```

```
>color(5); plotLine(l); plotLine(g); color(1);
>plotPoint(P,“P”);
>plotCircle(circleWithCenter(P,r),“Lingkaran dalam segitiga
ABC”):
```



Gambar 12.297 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-005.png

```
>r=norm(P-projectToLine(P,lineThrough(A,B)))

1.52896119631

>plotCircle(circleWithCenter(P,r),“Lingkaran dalam segitiga
ABC”):
```



Gambar 12.298 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-006.png

CONTOH 2: GEOMETRI SMBOLIK

Kita dapat menghitung geometri eksak dan simbolik menggunakan Maxima.

File `geometri.e` menyediakan fungsi-fungsi yang sama (dan lebih banyak lagi) di Maxima. Namun, kita dapat menggunakan komputasi simbolik sekarang.

```
>A &= [1,0]; B &= [0,1]; C &= [2,2]; // menentukan tiga titik A, B, C
```

Fungsi untuk garis dan lingkaran bekerja seperti fungsi Euler, tetapi menyediakan komputasi simbolis.

```
>c &= lineThrough(B,C) // c=BC
```

$$[-1, 2, 2]$$

Kita bisa mendapatkan persamaan untuk sebuah garis dengan mudah.

```
>$getLineEquation(c,x,y), $solve(%,y) | expand // persamaan garis c
```

$$2y - x = 2$$

$$\left[y = \frac{x}{2} + 1 \right]$$

```
>$getLineEquation(lineThrough([x1,y1],[x2,y2]),x,y), $solve(%,y) // persamaan garis melalui (x1, y1) dan (x2, y2)
```

$$x(y_1 - y_2) + (x_2 - x_1)y = x_1(y_1 - y_2) + (x_2 - x_1)y_1$$

$$\left[y = \frac{-(x_1 - x)y_2 - (x - x_2)y_1}{x_2 - x_1} \right]$$

```
>$getLineEquation(lineThrough(A,[x1,y1]),x,y) // persamaan garis melalui A dan (x1, y1)
```

$$(x_1 - 1)y - xy_1 = -y_1$$

>h &= perpendicular(A,lineThrough(B,C)) // h melalui A tegak lurus BC

$$[2, 1, 2]$$

>Q &= lineIntersection(c,h) // Q titik potong garis c=BC dan h

$$\begin{bmatrix} 2 & 6 \\ -\frac{2}{5} & -\frac{6}{5} \end{bmatrix}$$

>\$projectToLine(A,lineThrough(B,C)) // proyeksi A pada BC

$$\begin{bmatrix} \frac{2}{5} & \frac{6}{5} \end{bmatrix}$$

>\$distance(A,Q) // jarak AQ

$$\frac{3}{\sqrt{5}}$$

>cc &= circleThrough(A,B,C); \$cc // (titik pusat dan jari-jari) lingkaran melalui A, B, C

$$\left[\frac{7}{6}, \frac{7}{6}, \frac{5}{3\sqrt{2}} \right]$$

>r&=getCircleRadius(cc); \$r , \$float(r) // tampilkan nilai jari-jari

$$\frac{5}{3\sqrt{2}}$$

1.178511301977579

>\$computeAngle(A,C,B) // nilai <ACB

$$\arccos\left(\frac{4}{5}\right)$$

>\$solve(getLineEquation(angleBisector(A,C,B),x,y),y)[1] // persamaan garis bagi <ACB

$$y = x$$

```
>P &= lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A));
$P // titik potong 2 garis bagi sudut
```

$$\left[\frac{\sqrt{2}\sqrt{5}+2}{6}, \frac{\sqrt{2}\sqrt{5}+2}{6} \right]$$

```
>P() // hasilnya sama dengan perhitungan sebelumnya
```

```
[0.86038, 0.86038]
```

Garis dan Lingkaran yang Berpotongan

Tentu saja, kita juga bisa memotong garis dengan lingkaran, dan lingkaran dengan lingkaran.

```
>A &:= [1,0]; c:=circleWithCenter(A,4);
>B &:= [1,2]; C &:= [2,1]; l:=lineThrough(B,C);
>setPlotRange(5); plotCircle(c); plotLine(l);
```

Perpotongan garis dengan lingkaran menghasilkan dua titik dan jumlah titik perpotongan.

```
>{P1,P2,f}=lineCircleIntersections(l,c);
>P1, P2, f
```

```
[4.64575, -1.64575]
[-0.645751, 3.64575]
2
```

```
>plotPoint(P1); plotPoint(P2):
```

Hal yang sama 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
```



Gambar 12.299 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-020.png

[1, 1, 3]

>\$lineCircleIntersections(l,c) | radcan, // titik potong lingkaran c dan garis l

$$\left[\left[\sqrt{7} + 2, 1 - \sqrt{7} \right], \left[2 - \sqrt{7}, \sqrt{7} + 1 \right] \right]$$

Akan ditunjukkan bahwa sudut-sudut yang menghadap busur yang sama adalah sama besar.

>C=A+normalize([-2,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);

>degprint(computeAngle(P1,C,P2))

69°17'42.68''

>C=A+normalize([-4,-3])*4; plotPoint(C); plotSegment(P1,C); plotSegment(P2,C);

>degprint(computeAngle(P1,C,P2))

69°17'42.68''

>insimg;

Garis Sumbu

Berikut adalah langkah-langkah menggambar garis sumbu ruas garis AB:

1. Gambar lingkaran dengan pusat A melalui B.

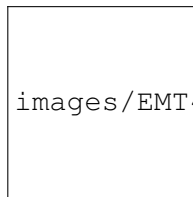


images/EMT4Geometry_Shintia Nurchol

Gambar 12.300 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-022.png

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/EMT4Geometry_Shintia Nurchol

Gambar 12.301 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-023.png

Selanjutnya, kami melakukan hal yang sama di Maxima dengan koordinat umum.

```
>A &= [a1,a2]; B &= [b1,b2];
>c1 &= circleWithCenter(A,distance(A,B));
```

```
>c2 &= circleWithCenter(B,distance(A,B));
>P &= circleCircleIntersections(c1,c2); P1 &= P[1]; P2 &=
P[2];
```

Persamaan untuk persimpangan cukup rumit. Tetapi kita dapat menyederhanakannya, jika kita menyelesaikan untuk y.

```
>g &= getLineEquation(lineThrough(P1,P2),x,y);
>$solve(g,y)
```

$$\left[y = \frac{-(2b_1 - 2a_1)x + b_2^2 + b_1^2 - a_2^2 - a_1^2}{2b_2 - 2a_2} \right]$$

Ini memang sama dengan tegak lurus tengah, yang dihitung dengan cara yang sama sekali berbeda.

```
>$solve(getLineEquation(middlePerpendicular(A,B),x,y),y)
```

$$\left[y = \frac{-(2b_1 - 2a_1)x + b_2^2 + b_1^2 - a_2^2 - a_1^2}{2b_2 - 2a_2} \right]$$

```
>h &=getLineEquation(lineThrough(A,B),x,y);
>$solve(h,y)
```

$$\left[y = \frac{(b_2 - a_2)x - a_1b_2 + a_2b_1}{b_1 - a_1} \right]$$

Contoh 3: Rumus Heron

Rumus Heron menyatakan bahwa luas segitiga dengan panjang sisi-sisi a, b dan c adalah:

$$L = \sqrt{s(s-a)(s-b)(s-c)} \quad \text{dengan } s = (a+b+c)/2,$$

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):
```



Gambar 12.302 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-031.png

```
>&assume(a>0); sol &= solve([x^2+y^2=b^2,(x-a)^2+y^2=c^2],[x,y])
```

```
[ ]
```

Ekstrak solusi y.

```
>ysol &= y with sol[2][2]; $'y=sqrt(factor(ysol^2))
```

Maxima said:

```
part: invalid index of list or matrix.
```

```
-- an error. To debug this try: debugmode(true);
```

Error in:

```
ysol &= y with sol[2][2]; $'y=sqrt(factor(ysol^2))
```

Kita mendapatkan rumus Heron.

```
>function H(a,b,c) &= sqrt(factor((ysol*a/2)^2)); $'H(a,b,c)=H(a,b,c)
```

$$H(a,b,[1,0,4]) = \frac{a |ysol|}{2}$$

```
>$'Luas=H(2,5,6) // luas segitiga dengan panjang sisi-sisi 2, 5, 6
```

$$Luas = |ysol|$$

Tentu saja, setiap segitiga persegi panjang adalah kasus yang terkenal.

```
>H(3,4,5) //luas segitiga siku-siku dengan panjang sisi 3, 4, 5
```

Variable or function ysol not found.

Try "trace errors" to inspect local variables after errors

H:

```
useglobal; return a*abs(ysol)/2
```

Error in:

```
H(3,4,5) //luas segitiga siku-siku dengan panjang sisi 3,  
^
```

Dan juga jelas, bahwa ini adalah segitiga dengan luas maksimal dan kedua sisi 3 dan 4.

```
>aspect (1.5); plot2d(&H(3,4,x),1,7): // Kurva luas segitiga  
sengan panjang sisi 3, 4, x (1<= x <=7)
```

Variable or function ysol not found.

Error in expression: 3*abs(ysol)/2

%ploteval:

```
y0=f$(x[1],args());
```

adaptiveevalone:

```
s=%ploteval(g$,t,args());
```

Try "trace errors" to inspect local variables after errors

plot2d:

```
dw/n,dw/n^2,dw/n,auto;args());
```

Kasus umum juga bisa digunakan.

```
>$solve(diff(H(a,b,c)^2,c)=0,c)
```

Maxima said:

```
diff: second argument must be a variable; found [1,0,4]
-- an error. To debug this try: debugmode(true);
```

Error in:

```
$solve(diff(H(a,b,c)^2,c)=0,c) ...
```

Sekarang mari kita cari himpunan semua titik di mana $b+c=d$ untuk suatu konstanta d . Sudah diketahui bahwa ini adalah sebuah elips.

```
>s1 &= subst(d-c,b,sol[2]); $s1
```

Maxima said:

```
part: invalid index of list or matrix.
```

```
-- an error. To debug this try: debugmode(true);
```

Error in:

```
s1 &= subst(d-c,b,sol[2]); $s1 ...
```

Dan membuat fungsi-fungsi ini.

```
>function fx(a,c,d) &= rhs(s1[1]); $fx(a,c,d), function fy(a,c,d)
&= rhs(s1[2]); $fy(a,c,d)
```

0

0

Sekarang kita dapat menggambar himpunan tersebut. Sisi b bervariasi dari 1 hingga 4. Sudah diketahui bahwa kita mendapatkan sebuah elips.

```
>aspect(1); plot2d(&fx(3,x,5),&fy(3,x,5),xmin=1,xmax=4,square=1):
```

Kita dapat memeriksa persamaan umum untuk elips ini, yaitu

$$\frac{(x - x_m)^2}{u^2} + \frac{(y - y_m)^2}{v^2} = 1,$$

di mana (x_m, y_m) adalah pusat, dan u serta v adalah setengah sumbu.

```
>$ratsimp((fx(a,c,d)-a/2)^2/u^2+fy(a,c,d)^2/v^2 with [u=d/2,v=sqrt(d^2-a^2)/2])
```




Gambar 12.303 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-036.png

$$\frac{a^2}{d^2}$$

Kita melihat bahwa tinggi dan luas segitiga adalah maksimal untuk $x=0$. Dengan demikian, luas segitiga dengan $a+b+c=d$ adalah maksimal, jika segitiga tersebut sama sisi. Kita ingin membuktikannya secara analitis.

```
>eqns &= [diff(H(a,b,d-(a+b))2,a)=0,diff(H(a,b,d-(a+b))2,b)=0];
$eqns
```

$$\left[\frac{a^2}{2} = 0, 0 = 0 \right]$$

Kita mendapatkan beberapa minima, yang termasuk dalam segitiga dengan satu sisi 0, dan solusi $a = b = c = d / 3$.

```
>$solve(eqns,[a,b])
```

$$[[a = 0, b = \%r_1]]$$

Ada juga metode Lagrange, yang memaksimalkan $H(a,b,c)^2$ sehubungan dengan $a+b+c=d$.

```
>&solve([diff(H(a,b,c)2,a)=la,diff(H(a,b,c)2,b)=la, ...
> diff(H(a,b,c)2,c)=la,a+b+c=d],[a,b,c,la])
```

Maxima said:

```
diff: second argument must be a variable; found [1,0,4]
-- an error. To debug this try: debugmode(true);
```

Error in:

```
... la,      diff(H(a,b,c)^2,c)=la,a+b+c=d],[a,b,c,la])
```

Kita bisa membuat plot situasi

Pertama-tama, tetapkan titik-titik di Maxima.

```
>A &= at([x,y],sol[2]); $A
```

Maxima said:

```
part: invalid index of list or matrix.
```

```
-- an error. To debug this try: debugmode(true);
```

Error in:

```
A &= at([x,y],sol[2]); $A ...
      ^
```

```
>B &= [0,0]; $B, C &= [a,0]; $C
```

```
[0,0]
```

```
[a,0]
```

Kemudian, tetapkan kisaran plot, dan plot titik-titiknya.

```
>setPlotRange(0,5,-2,3); ...
```

```
> a=4; b=3; c=2; ...
```

```
> plotPoint(mxmeval("B"),"B"); plotPoint(mxmeval("C"),"C"); ...
```

```
> plotPoint(mxmeval("A"),"A");
```

Variable a1 not found!

Use global variables or parameters for string evaluation.

Error in Evaluate, superfluous characters found.

Try "trace errors" to inspect local variables after evaluation.

mxmeval:

```
return evaluate(mxm(s));
```

Error in:

```
... otPoint(mxmeval("C"),"C"); plotPoint(mxmeval("A"));
```

Plot segmen-segmen tersebut.

```
>plotSegment(mxmeval("A"),mxmeval("C")); ...
> plotSegment(mxmeval("B"),mxmeval("C")); ...
> plotSegment(mxmeval("B"),mxmeval("A")):
```

Variable a1 not found!

Use global variables or parameters for string evaluation.
 Error in Evaluate, superfluous characters found.
 Try "trace errors" to inspect local variables after errors
 mxmeval:

```
return evaluate (mxm(s));
```

Error in:

```
plotSegment (mxmeval ("A"),mxmeval ("C")); plotSegment (mxmeval ("B"),mxmeval ("A"));
```

Hitung garis tegak lurus tengah dalam Maxima.

```
>h &= middlePerpendicular(A,B); g &= middlePerpendicular(B,C);
```

Dan bagian tengah lingkaran.

```
>U &= lineIntersection(h,g);
```

Kita mendapatkan rumus untuk jari-jari lingkaran.

```
>&assume(a>0,b>0,c>0); $distance(U,B) | radcan
```

$$\frac{\sqrt{a_2^2 + a_1^2} \sqrt{a_2^2 + a_1^2 - 2 a a_1 + a^2}}{2 |a_2|}$$

Mari kita tambahkan ini ke dalam plot.

```
>plotPoint(U()); ...
```

```
> plotCircle(circleWithCenter(mxmeval("U"),mxmeval("distance(U,C)"))):
```

Variable a2 not found!

Use global variables or parameters for string evaluation.
 Error in ^

```
Error in expression: [a/2, (a2^2+a1^2-a*a1)/(2*a2)]
```

Error in:

```
plotPoint (U()); plotCircle (circleWithCenter (mxmeval ("U"),mxmeval ("distance(U,C)")));
```

Dengan menggunakan geometri, kami memperoleh rumus sederhana

$$\frac{a}{\sin(\alpha)} = 2r$$

untuk radius. Kita bisa mengecek, apakah hal ini benar dengan Maxima. Maxima akan memperhitungkannya hanya jika kita mengkuadratkannya.

```
>$c^2/sin(computeAngle(A,B,C))2 | factor
```

$$\left[\frac{a_2^2 + a_1^2}{a_2^2}, 0, \frac{16 (a_2^2 + a_1^2)}{a_2^2} \right]$$

Contoh 4: Garis Euler dan Parabola

Garis Euler adalah garis yang ditentukan dari segitiga apa pun yang tidak sama sisi. Garis ini merupakan garis tengah segitiga, dan melewati beberapa titik penting yang ditentukan dari segitiga, termasuk ortosentrum, circumcentrum, centroid, titik Exeter, dan pusat lingkaran sembilan titik segitiga.

Sebagai demonstrasi, kami menghitung dan memplot garis Euler dalam sebuah segitiga.

Pertama, kita mendefinisikan sudut-sudut segitiga dalam Euler. Kami menggunakan definisi, yang terlihat dalam ekspresi simbolis.

```
>A::=[-1,-1]; B::=[2,0]; C::=[1,2];
```

Untuk memplot objek geometris, kita menyiapkan area plot, dan menambahkan titik-titiknya. Semua plot objek geometris ditambahkan ke plot saat ini.

```
>setPlotRange(3); plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C");
```

Kita juga bisa menambahkan sisi-sisi segitiga.

```
>plotSegment(A,B,""); plotSegment(B,C,""); plotSegment(C,A,"");
```

Berikut ini adalah luas area segitiga, dengan menggunakan rumus determinan. Tentu saja, kita harus mengambil nilai absolut dari hasil ini.

```
>$areaTriangle(A,B,C)
```

$$-\frac{7}{2}$$

Kita dapat menghitung koefisien dari sisi c.



Gambar 12.304 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-046.png

```
>c &= lineThrough(A,B)
```

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

Dan juga mendapatkan formula 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 tersebut akan menghasilkan jarak positif ke garis.

```
>$getHesseForm(c,x,y,C), $at(%, [x=C[1],y=C[2]])
```

$$\frac{3y - x + 2}{\sqrt{10}}$$

$$\frac{7}{\sqrt{10}}$$

Sekarang kita menghitung keliling ABC.

```
>LL &= circleThrough(A,B,C); $getCircleEquation(LL,x,y)
```

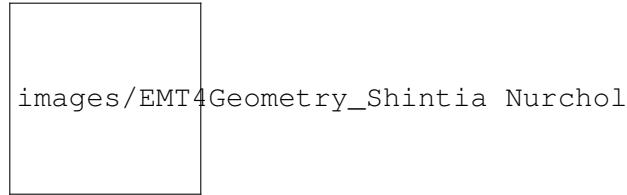
$$\left(y - \frac{5}{14}\right)^2 + \left(x - \frac{3}{14}\right)^2 = \frac{325}{98}$$

```
>O &= getCircleCenter(LL); $O
```

$$\left[\frac{3}{14}, \frac{5}{14}\right]$$

Plot lingkaran dan pusatnya. Cu dan U adalah simbolik. Kami mengevaluasi ekspresi ini untuk Euler.

```
>plotCircle(LL()); plotPoint(O(),“O”):
```



Gambar 12.305 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-053.png

Kita dapat menghitung perpotongan ketinggian di ABC (pusat ortosentrum) secara numerik dengan perintah berikut ini.

```
>H &= lineIntersection(perpendicular(A,lineThrough(C,B)),...
> perpendicular(B,lineThrough(A,C))); $H
```

$$\left[\frac{11}{7}, \frac{2}{7} \right]$$

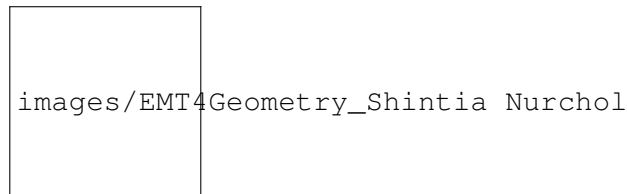
Sekarang kita dapat menghitung garis Euler dari segitiga tersebut.

```
>el &= lineThrough(H,O); $getLineEquation(el,x,y)
```

$$-\frac{19y}{14} - \frac{x}{14} = -\frac{1}{2}$$

Tambahkan ke plot kami.

```
>plotPoint(H(),“H”); plotLine(el(),“Garis Euler”):
```



Gambar 12.306 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-056.png

Pusat gravitasi harus berada pada garis ini.

>M &= (A+B+C)/3; \$getLineEquation(el,x,y) with

$$-\frac{1}{2} = -\frac{1}{2}$$

>plotPoint(M(),"M"): // titik berat



Gambar 12.307 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-058.png

Teori mengatakan bahwa $MH = 2 \cdot MO$. Kita perlu menyederhanakan dengan radcan untuk mencapai hal ini.

>\$distance(M,H)/distance(M,O)|radcan

2

Fungsi-fungsi ini juga mencakup fungsi untuk sudut.

>\$computeAngle(A,C,B), degprint(%)

$$\arccos\left(\frac{4}{\sqrt{5}\sqrt{13}}\right)$$

60°15'18.43''

Persamaan untuk bagian tengah lingkaran tidak terlalu bagus.

>Q &= lineIntersection(angleBisector(A,C,B),angleBisector(C,B,A))|radcan;
\$Q

$$\left[\frac{\left(2^{\frac{3}{2}} + 1\right) \sqrt{5} \sqrt{13} - 15 \sqrt{2} + 3}{14}, \frac{(\sqrt{2} - 3) \sqrt{5} \sqrt{13} + 5 \cdot 2^{\frac{3}{2}} + 5}{14} \right]$$

Mari kita hitung juga ekspresi untuk jari-jari lingkaran yang tertulis.

```
>r &= distance(Q,projectToLine(Q,lineThrough(A,B)))/ratsimp;
$r
```

$$\frac{\sqrt{(-41\sqrt{2}-31)\sqrt{5}\sqrt{13}+115\sqrt{2}+614}}{7\sqrt{2}}$$

```
>LD &= circleWithCenter(Q,r); // Lingkaran dalam
Mari kita tambahkan ini ke dalam plot.
>color(5); plotCircle(LD());
```



Gambar 12.308 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-063.png

Parabola

Selanjutnya akan dicari persamaan tempat kedudukan titik-titik yang berjarak sama ke titik C dan ke garis AB.

```
>p &= getHesseForm(lineThrough(A,B),x,y,C)-distance([x,y],C);
$p='0
```

$$\frac{3y-x+2}{\sqrt{10}} - \sqrt{(2-y)^2 + (1-x)^2} = 0$$

Persamaan tersebut dapat digambar menjadi satu dengan gambar sebelumnya.

```
>plot2d(p,level=0,add=1,contourcolor=6):
```

Ini seharusnya merupakan suatu fungsi, tetapi solver default Maxima hanya bisa menemukan solusinya jika kita mengkuadratkan persamaannya. Akibatnya, kita mendapatkan solusi palsu.

```
>akar &= solve(getHesseForm(lineThrough(A,B),x,y,C)^2-distance([x,y],C)^2,y
```

$$[y = -3x - \sqrt{70}\sqrt{9-2x} + 26,$$



images/EMT4Geometry_Shintia Nurcholisa_2

Gambar 12.309 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-065.png

$$y = -3x + \sqrt{70} \sqrt{9 - 2x} + 26$$

Solusi pertama adalah

maxima: akar[1]

Menambahkan solusi pertama ke dalam plot menunjukkan, bahwa itu memang jalur yang kita cari. Teori mengatakan bahwa itu adalah parabola yang diputar.

>plot2d(&rhs(akar[1]),add=1):



images/EMT4Geometry_Shintia Nurcholisa_2

Gambar 12.310 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-066.png

>function g(x) &= rhs(akar[1]); \$'g(x)= g(x)// fungsi yang mendefinisikan kurva di atas

$$g(x) = -3x - \sqrt{70} \sqrt{9 - 2x} + 26$$

>T &=[-1, g(-1)]; // ambil sebarang titik pada kurva tersebut

>dTC &= distance(T,C); \$fullratsimp(dTC), \$float(%) // jarak T ke C

$$\sqrt{1503 - 54\sqrt{11}\sqrt{70}}$$

2.135605779339061

>U &= projectToLine(T,lineThrough(A,B)); \$U // proyeksi T pada garis AB

$$\left[\frac{80 - 3\sqrt{11}\sqrt{70}}{10}, \frac{20 - \sqrt{11}\sqrt{70}}{10} \right]$$

>dU2AB &= distance(T,U); \$fullratsimp(dU2AB), \$float(%) // jarak T ke AB

$$\sqrt{1503 - 54\sqrt{11}\sqrt{70}}$$

2.135605779339061

Ternyata jarak T ke C sama dengan jarak T ke AB. Coba Anda pilih titik T yang lain dan ulangi perhitungan-perhitungan di atas untuk menunjukkan bahwa hasilnya juga sama.

CONTOH 5: TRIGONOMETRI RASIONAL

Hal ini terinspirasi dari sebuah ceramah N.J. Wildberger. Dalam bukunya “Proporsi Ilahi”, Wildberger mengusulkan untuk mengganti gagasan klasik tentang jarak dan sudut dengan kuadransi dan penyebaran. Dengan menggunakan ini, memang memungkinkan untuk menghindari fungsi trigonometri dalam banyak contoh, dan tetap “rasional”.

Berikut ini, saya akan memperkenalkan konsep-konsep tersebut, dan memecahkan beberapa masalah. Saya menggunakan komputasi simbolik Maxima di sini, yang menyembunyikan keuntungan utama dari trigonometri rasional yaitu komputasi dapat dilakukan dengan kertas dan pensil saja. Anda dipersilakan untuk memeriksa hasilnya tanpa komputer.

Intinya adalah bahwa komputasi rasional simbolik sering kali memberikan hasil yang sederhana. Sebaliknya, trigonometri klasik menghasilkan hasil trigonometri yang rumit, yang dievaluasi dengan pendekatan numerik saja.

```
>load geometry;
```

Untuk pengenalan pertama, kita menggunakan segitiga persegi panjang dengan proporsi Mesir yang terkenal 3, 4, dan 5. Perintah berikut ini adalah perintah Euler untuk memplot geometri bidang yang terdapat pada file Euler “geometry.e”.

```
>C&:=[0,0]; A&:=[4,0]; B&:=[0,3]; ...  
> setPlotRange(-1,5,-1,5); ...  
> plotPoint(A,“A”); plotPoint(B,“B”); plotPoint(C,“C”); ...  
> plotSegment(B,A,“c”); plotSegment(A,C,“b”); plotSegment(C,B,“a”);  
...  
> insimg(30);  
tentu,
```

$$\sin(w_a) = \frac{a}{c},$$

di mana w_a adalah sudut di A. Cara biasa untuk menghitung sudut ini, adalah dengan mengambil kebalikan dari fungsi sinus. Hasilnya adalah



Gambar 12.311 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-073.png

sudut yang tidak dapat dicerna, yang hanya dapat dicetak kira-kira.

```
>wa := arcsin(3/5); degprint(wa)
```

36°52'11.63''

Trigonometri rasional mencoba menghindari hal ini.

Gagasan pertama trigonometri rasional adalah kuadrat, yang menggantikan jarak. Sebenarnya, ini hanyalah jarak yang dikuadratkan. Berikut ini, a, b, dan c menunjukkan kuadran sisi-sisinya.

Teorema Pythagoras menjadi $a^2 + b^2 = c^2$.

```
>a^2 + b^2 = c^2; a = 3; b = 4; c = 5; a+b=c
```

$$25 = 25$$

Gagasan kedua dari trigonometri rasional adalah penyebaran. Penyebaran mengukur bukaan di antara garis-garis. Ini adalah 0, jika garis-garisnya sejajar, dan 1, jika garis-garisnya persegi panjang. Ini adalah kuadrat dari sinus sudut antara dua garis.

Penyebaran garis AB dan AC pada gambar di atas didefinisikan sebagai

$$s_a = \sin(\alpha)^2 = \frac{a}{c},$$

di mana a dan c adalah kuadran dari segitiga persegi panjang dengan satu sudut di A.

```
>sa = a/c; $sa
```

$$\frac{9}{25}$$

Tentu saja, hal ini lebih mudah dihitung daripada sudut. Tetapi Anda kehilangan sifat bahwa sudut dapat ditambahkan dengan mudah.

Tentu saja, kita bisa mengonversi nilai perkiraan kita untuk sudut wa ke sprad, dan mencetaknya sebagai pecahan.

```
>fracprint(sin(wa)^2)
```

9/25

Hukum kosinus trgonometri klasik diterjemahkan ke dalam “hukum silang” berikut ini.

$$(c + b - a)^2 = 4bc(1 - s_a)$$

Di sini a, b, dan c adalah kuadran dari sisi-sisi segitiga, dan sa adalah penyebaran di sudut A. Sisi a, seperti biasa, berlawanan dengan sudut A.

Hukum-hukum ini diimplementasikan dalam file geometri.e yang kita masukkan ke dalam Euler.

```
>$crosslaw(aa,bb,cc,saa)
```

$$\left[\left(bb - aa + \frac{7}{6} \right)^2, \left(bb - aa + \frac{7}{6} \right)^2, \left(bb - aa + \frac{5}{3\sqrt{2}} \right)^2 \right] = \left[\frac{14 bb (1 -}{3} \right.$$

Dalam kasus kami, kami mendapatkan

```
>$crosslaw(a,b,c,sa)
```

$$1024 = 1024$$

Mari kita gunakan crosslaw ini untuk mencari sebaran di A. Untuk melakukannya, kita buat crosslaw untuk kuadran a, b, dan c, dan selesaikan untuk sebaran sa yang tidak diketahui.

Anda bisa melakukan ini dengan tangan dengan mudah, tapi saya menggunakan Maxima. Tentu saja, kita mendapatkan hasil yang sudah kita dapatkan.

```
>$crosslaw(a,b,c,x), $solve(%,x)
```

$$1024 = 1600 (1 - x)$$

$$\left[x = \frac{9}{25} \right]$$

Kita sudah mengetahui hal ini. Definisi penyebaran adalah kasus khusus dari crosslaw.

Kita juga dapat menyelesaikannya untuk a, b, c secara umum. Hasilnya adalah sebuah rumus yang menghitung penyebaran sudut sebuah segitiga dengan kuadran ketiga sisinya.

```
>$solve(crosslaw(aa,bb,cc,x),x)
```

$$\left[\left[\frac{168 bb x + 36 bb^2 + (-72 aa - 84) bb + 36 aa^2 - 84 aa + 49}{36}, \frac{168 bb x + 36 bb^2}{36} \right] \right]$$

Kita dapat membuat sebuah fungsi dari hasil tersebut. Fungsi seperti itu sudah didefinisikan dalam file geometry.e dari Euler.

```
>$spread(a,b,c)
```

$$\frac{9}{25}$$

Sebagai contoh, kita dapat menggunakannya untuk menghitung sudut segitiga dengan sisi

$$a, \quad a, \quad \frac{4a}{7}$$

Hasilnya adalah rasional, yang tidak mudah didapat jika kita menggunakan trigonometri klasik.

```
>$spread(a,a,4*a/7)
```

$$\frac{6}{7}$$

Ini adalah sudut dalam derajat.

```
>degprint(arcsin(sqrt(6/7)))
```

$$67^{\circ} 47' 32.44''$$

Contoh Lain

Sekarang, mari kita coba contoh yang lebih lanjut.

Kita tentukan tiga sudut segitiga sebagai berikut.

```
> A&:=[1,2]; B&:=[4,3]; C&:=[0,4]; ...  
> setPlotRange(-1,5,1,7); ...  
> plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...  
> plotSegment(B,A,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a");  
...  
> insimg;
```



Gambar 12.312 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-086.png

Dengan menggunakan Pythagoras, mudah untuk menghitung jarak antara dua titik. Pertama-tama saya menggunakan jarak fungsi dari file Euler untuk geometri. Jarak fungsi menggunakan geometri klasik.

```
> $distance(A,B)
```

$$\sqrt{10}$$

Euler juga memiliki fungsi untuk kuadranan antara dua titik.

Pada contoh berikut, karena $c+b$ bukan a , maka segitiga tersebut tidak berbentuk persegi panjang.

```
> c &= quad(A,B); $c, b &= quad(A,C); $b, a &= quad(B,C);  
$a,
```

10

5

17

33

Pertama, mari kita menghitung sudut tradisional. Fungsi computeAngle menggunakan metode yang biasa berdasarkan hasil kali titik dari dua vektor. Hasilnya adalah beberapa perkiraan titik mengambang.

$$A = \langle 1, 2 \rangle \quad B = \langle 4, 3 \rangle, \quad C = \langle 0, 4 \rangle$$

$$\mathbf{a} = C - B = \langle -4, 1 \rangle, \quad \mathbf{c} = A - B = \langle -3, -1 \rangle, \quad \beta = \angle ABC$$

$$\mathbf{a} \cdot \mathbf{c} = |\mathbf{a}| \cdot |\mathbf{c}| \cos \beta$$

$$\cos \angle ABC = \cos \beta = \frac{\mathbf{a} \cdot \mathbf{c}}{|\mathbf{a}| \cdot |\mathbf{c}|} = \frac{12 - 1}{\sqrt{17} \sqrt{10}} = \frac{11}{\sqrt{17} \sqrt{10}}$$

>wb &= computeAngle(A,B,C); \$wb, \$(wb/pi*180)()

$$\arccos\left(\frac{11}{\sqrt{10}\sqrt{17}}\right)$$

32.4711922908

Dengan menggunakan pensil dan kertas, kita dapat melakukan hal yang sama dengan hukum silang. Kita masukkan kuadran a, b, dan c ke dalam hukum silang dan selesaikan untuk x.

$$>\$crosslaw(a,b,c,x), \$solve(\%,x), //(b+c-a)^{\wedge}4b.c(1-x)$$

$$4 = 200 (1 - x)$$

$$\left[x = \frac{49}{50} \right]$$

Itulah yang dilakukan oleh fungsi spread yang didefinisikan dalam “geometry.e”.

$$>sb \&= spread(b,a,c); \$sb$$

$$\frac{49}{170}$$

Maxima mendapatkan hasil yang sama dengan menggunakan trigonometri biasa, jika kita memaksakannya. Ia menyelesaikan suku $\sin(\arccos(\dots))$ menjadi hasil pecahan. Sebagian besar siswa tidak dapat melakukan ini.

$$>\$sin(computeAngle(A,B,C))^2$$

$$\frac{49}{170}$$

Setelah kita memiliki penyebaran di B, kita dapat menghitung tinggi ha di sisi a. Ingatlah bahwa

$$s_b = \frac{h_a}{c}$$

menurut definisi.

$$h_a = c \cdot s_b$$

$$\frac{49}{17}$$

Gambar berikut ini dibuat dengan program geometri C.a.R., yang dapat menggambar kuadran dan penyebaran.

image: (20) Rational_Geometry_CaR.png

Menurut definisi, panjang ha adalah akar kuadrat dari kuadrannya.

$$h_a = \sqrt{a \cdot s_a}$$

$$\frac{7}{\sqrt{17}}$$

Sekarang kita dapat menghitung luas segitiga. Jangan lupa, bahwa kita berurusan dengan kuadran!

$$L = \frac{1}{2} \cdot a \cdot h_a$$

$$\frac{7}{2}$$

Rumus penentu yang biasa menghasilkan hasil yang sama.

$$L = \frac{1}{4} \sqrt{(a+b+c)(a+b-c)(a+c-b)(b+c-a)}$$

$$\frac{7}{2}$$

Rumus The Heron

sekarang, mari kita selesaikan masalah ini secara umum!

$$L = \frac{1}{4} \sqrt{(a+b+c)(a+b-c)(a+c-b)(b+c-a)}$$

Pertama-tama kita menghitung penyebaran di B untuk segitiga dengan sisi a, b, dan c. Kemudian kita menghitung luas kuadrat

(“quadrea?”), memfaktorkannya dengan Maxima, dan kita mendapatkan rumus Heron yang terkenal.

emang, hal ini sulit dilakukan dengan pensil dan kertas.

`>$spread(b^2*c2,a^2), $factor(%*c^2*a2/4)`

$$\frac{-c^4 - (-2b^2 - 2a^2)c^2 - b^4 + 2a^2b^2 - a^4}{4a^2c^2}$$

$$\frac{(-c + b + a)(c - b + a)(c + b - a)(c + b + a)}{16}$$

Aturan Triple Spread

erugian dari spread adalah bahwa mereka tidak lagi hanya menambahkan sudut seperti.

Namun, tiga spread dari sebuah segitiga memenuhi aturan “triple spread” berikut ini.

`>&remvalue(sa,sb,sc); $triplespread(sa,sb,sc)`

$$(sc + sb + sa)^2 = 2(sc^2 + sb^2 + sa^2) + 4sa sb sc$$

Aturan ini berlaku untuk tiga sudut yang berjumlah 180° .

$$\alpha + \beta + \gamma = \pi$$

Karena spread dari

$$\alpha, \pi - \alpha$$

sama, aturan triple spread juga benar, jika

$$\alpha + \beta = \gamma$$

Karena penyebaran sudut negatifnya sama, aturan penyebaran tiga kali lipat juga berlaku, jika

$$\alpha + \beta + \gamma = 0$$

Contohnya, kita bisa menghitung penyebaran sudut 60° . Hasilnya adalah $3/4$. Namun, persamaan ini memiliki solusi kedua, di mana semua penyebarannya adalah 0.

```
>$solve(triplespread(x,x,x),x)
```

$$\left[x = \frac{3}{4}, x = 0 \right]$$

Penyebaran 90° jelas adalah 1. Jika dua sudut ditambahkan ke 90° , penyebarannya akan menyelesaikan persamaan penyebaran tiga dengan a, b, 1. Dengan perhitungan berikut, kita mendapatkan $a + b = 1$.

```
>$triplespread(x,y,1), $solve(%,x)
```

$$(y + x + 1)^2 = 2 (y^2 + x^2 + 1) + 4 x y$$

$$[x = 1 - y]$$

Karena penyebaran 180° -t sama dengan penyebaran t, rumus penyebaran tiga kali lipat juga berlaku, jika satu sudut adalah jumlah atau selisih dari dua sudut lainnya.

Jadi kita dapat menemukan penyebaran sudut dua kali lipat. Perhatikan bahwa ada dua solusi lagi. Kita jadikan ini sebuah fungsi.

```
>$solve(triplespread(a,a,x),x), function doublespread(a) &=
factor(rhs( %[1] ))
```

$$[x = 4 a - 4 a^2, x = 0]$$

- 4 (a - 1) a

Pembagi Sudut

Ini adalah situasi yang sudah kita ketahui.

```
>C&:=[0,0]; A&:=[4,0]; B&:=[0,3]; ...
> setPlotRange(-1,5,-1,5); ...
> plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...
> plotSegment(B,A,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a");
...
> insimg;
```

Mari kita hitung panjang garis bagi sudut di A. Tetapi kita ingin menyelesaikannya untuk a, b, c secara umum.



images/EMT4Geometry_Shintia Nurcholisa_2

Gambar 12.313 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-116.png

```
>remvalue(a,b,c);
```

Jadi, pertama-tama kita menghitung penyebaran sudut yang dibelah dua di A, menggunakan rumus penyebaran tiga.

Masalah dengan rumus ini muncul lagi. Rumus ini memiliki dua solusi. Kita harus memilih salah satu yang benar. Solusi lainnya mengacu pada sudut terbagi dua 180° -wa.

```
>$triplespread(x,x,a/(a+b)), $solve(%x), sa2 &= rhs(%[1]);
$sa2
```

$$\left(2x + \frac{a}{b+a}\right)^2 = 2\left(2x^2 + \frac{a^2}{(b+a)^2}\right) + \frac{4ax^2}{b+a}$$

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

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

Mari kita periksa persegi panjang Mesir.

```
>$sa2 with [a=3^2,b=4^2]
```

$$\frac{1}{10}$$

Kita bisa mencetak sudut dalam Euler, setelah mentransfer penyebaran ke radian.

```
>wa2 := arcsin(sqrt(1/10)); degprint(wa2)
```

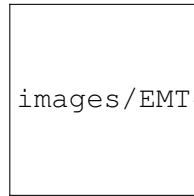
$18^\circ 26' 5.82''$

Titik P adalah perpotongan garis bagi sudut dengan sumbu y.

>P := [0,tan(wa2)*4]

[0, 1.33333]

>plotPoint(P,“P”); plotSegment(A,P):



Gambar 12.314 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-121.png

Mari kita periksa sudut-sudutnya dalam contoh spesifik kita.

>computeAngle(C,A,P), computeAngle(P,A,B)

0.321750554397

0.321750554397

Sekarang kita hitung panjang garis bagi AP.

Kita menggunakan teorema sinus dalam segitiga APC. Teorema ini menyatakan bahwa

$$\frac{BC}{\sin(w_a)} = \frac{AC}{\sin(w_b)} = \frac{AB}{\sin(w_c)}$$

berlaku dalam segitiga apa pun. Kuadratkan, ini diterjemahkan ke dalam apa yang disebut “hukum penyebaran”

$$\frac{a}{s_a} = \frac{b}{s_b} = \frac{c}{s_c}$$

di mana a, b, c menunjukkan kuadrannya.

Karena spread CPA adalah 1-sa2, kita mendapatkan bisa/1=b/(1-sa2) dan bisa menghitung bisa (kuadran dari pembagi sudut).

>&factor(ratsimp(b/(1-sa2))); bisa &= %; \$bisa

$$\frac{2b(b+a)}{\sqrt{b}\sqrt{b+a}+b+a}$$

Mari kita periksa rumus ini untuk nilai-nilai Mesir kita.

```
>sqrt(mxmeval("at(bisa,[a=32,b=42])")), distance(A,P)
```

```
4.21637021356
```

```
4.21637021356
```

Kita juga dapat menghitung P dengan menggunakan rumus penyebaran.

```
>py&=factor(ratsimp(sa2*bisa)); $py
```

$$-\frac{b\left(\sqrt{b}\sqrt{b+a}-b-a\right)}{\sqrt{b}\sqrt{b+a}+b+a}$$

Nilainya sama dengan yang kita dapatkan dengan rumus trigonometri.

```
>sqrt(mxmeval("at(py,[a=32,b=42])"))
```

```
1.33333333333
```

Sudut Akor

Lihatlah situasi berikut ini.

```
>setPlotRange(1.2); ...
> color(1); plotCircle(circleWithCenter([0,0],1)); ...
> A:=[cos(1),sin(1)]; B:=[cos(2),sin(2)]; C:=[cos(6),sin(6)]; ...
> plotPoint(A,"A"); plotPoint(B,"B"); plotPoint(C,"C"); ...
> color(3); plotSegment(A,B,"c"); plotSegment(A,C,"b"); plotSegment(C,B,"a");
...
> color(1); O:=[0,0]; plotPoint(O,"0"); ...
> plotSegment(A,O); plotSegment(B,O); plotSegment(C,O,"r"); ...
> insimg;
```

Kita dapat menggunakan Maxima untuk menyelesaikan rumus penyebaran tiga untuk sudut-sudut di pusat O untuk r. Dengan demikian kita mendapatkan rumus untuk jari-jari kuadrat dari pericircle dalam hal kuadran sisi-sisinya.



images/EMT4Geometry_Shintia Nurchol

Gambar 12.315 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-126.png

Kali ini, Maxima menghasilkan beberapa angka nol yang rumit, yang kita abaikan.

```
>&remvalue(a,b,c,r); // hapus nilai-nilai sebelumnya untuk
perhitungan baru
```

```
>rabc &= rhs(solve(triplespread(spread(b,r,r),spread(a,r,r),spread(c,r,r)),r)
$rabc
```

$$-\frac{a b c}{c^2 - 2 b c + a (-2 c - 2 b) + b^2 + a^2}$$

Kita dapat menjadikannya sebuah fungsi Euler.

```
>function periradius(a,b,c) &= rabc;
```

Mari kita periksa hasilnya untuk poin A, B, C.

```
>a:=quadrance(B,C); b:=quadrance(A,C); c:=quadrance(A,B);
```

Radiusnya memang 1.

```
>periradius(a,b,c)
```

1

Faktanya adalah, bahwa penyebaran CBA hanya bergantung pada b dan c. Ini adalah teorema sudut akor.

```
>$spread(b,a,c)*rabc | ratsimp
```

$$\frac{b}{4}$$

Faktanya, penyebarannya adalah b/(4r), dan kita melihat bahwa sudut chord b adalah setengah dari sudut tengah.

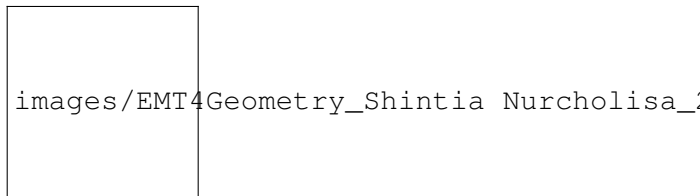
```
>$doublespread(b/(4*r))-spread(b,r,r) | ratsimp
```


Contoh 6: Jarak Minimal pada Bidang

Komentar awal

Fungsi yang, pada sebuah titik M pada bidang, menetapkan jarak AM antara titik tetap A dan M, memiliki garis-garis tingkat yang agak sederhana: lingkaran yang berpusat di A.

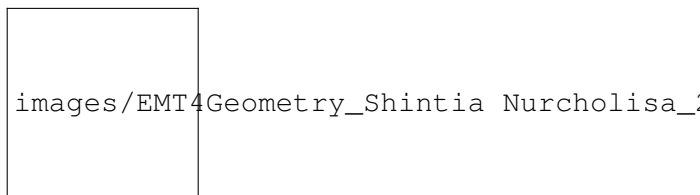
```
>&remvalue();
>A=[-1,-1];
>function d1(x,y):=sqrt((x-A[1])2+(y-A[2])2)
>fcontour("d1",xmin=-2,xmax=0,ymin=-2,ymax=0,hue=1,...
> title="If you see ellipses, please set your window square");
```



Gambar 12.316 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-130.png

dan grafiknya juga cukup sederhana: bagian atas kerucut:

```
>plot3d("d1",xmin=-2,xmax=0,ymin=-2,ymax=0):
```



Gambar 12.317 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-131.png

Tentu saja nilai minimum 0 diperoleh dalam A.

Dua titik

Sekarang kita lihat fungsi $MA+MB$ di mana A dan B adalah dua titik (tetap). Ini adalah “fakta yang terkenal” bahwa kurva tingkat adalah elips, titik fokusnya adalah A dan B ; kecuali AB minimum yang konstan pada segmen $[AB]$:

```
>B=[1,-1];  
>function d2(x,y):=d1(x,y)+sqrt((x-B[1])2+(y-B[2])2)  
>fcontour(“d2”,xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
```



Gambar 12.318 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-132.png

Grafiknya lebih menarik:

```
>plot3d(“d2”,xmin=-2,xmax=2,ymin=-3,ymax=1):
```



Gambar 12.319 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-133.png

Pembatasan pada garis (AB) lebih terkenal:

```
>plot2d(“abs(x+1)+abs(x-1)”,xmin=-3,xmax=3):
```



images/EMT4Geometry_Shintia Nurcholisa_2

Gambar 12.320 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-134.png

TIGA POIN

Sekarang, hal-hal menjadi kurang sederhana: Hal ini sedikit kurang dikenal bahwa $MA+MB+MC$ mencapai minimumnya pada satu titik di bidang, tetapi untuk menentukannya tidak sesederhana itu:

- 1) Jika salah satu sudut segitiga ABC lebih dari 120° (katakanlah di A), maka minimum dicapai pada titik ini (katakanlah AB+AC).

Contoh:

```
>C=[-4,1];  
>function d3(x,y):=d2(x,y)+sqrt((x-C[1])2+(y-C[2])2)  
>plot3d("d3",xmin=-5,xmax=3,ymin=-4,ymax=4);  
>insimg;
```



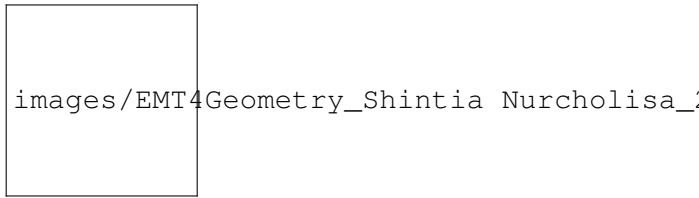
images/EMT4Geometry_Shintia Nurchol

Gambar 12.321 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-135.png

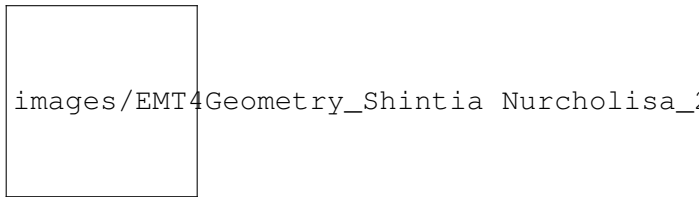
```
>fcontour("d3",xmin=-4,xmax=1,ymin=-2,ymax=2,hue=1,title="The  
minimum is on A");  
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);  
>insimg;
```

- 2) Tetapi jika semua sudut segitiga ABC kurang dari 120° , minimumnya adalah pada titik F di bagian dalam segitiga, yang merupakan satu-satunya titik yang melihat sisi-sisi ABC dengan sudut yang sama (masing-masing 120°):

```
>C=[-0.5,1];
```

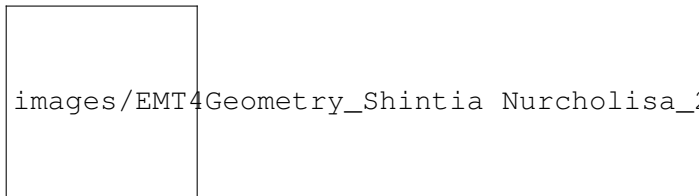


Gambar 12.322 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-136.png



Gambar 12.323 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-137.png

```
>plot3d("d3",xmin=-2,xmax=2,ymin=-2,ymax=2):  
>fcontour("d3",xmin=-2,xmax=2,ymin=-2,ymax=2,hue=1,title="The  
Fermat point");  
>P=(A_B_C_A)'; plot2d(P[1],P[2],add=1,color=12);  
>insimg;
```



Gambar 12.324 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-138.png

Merupakan kegiatan yang menarik untuk merealisasikan gambar di atas dengan perangkat lunak geometri; sebagai contoh, saya tahu sebuah perangkat lunak yang ditulis dalam bahasa Java yang memiliki instruksi “garis kontur”...

Semua hal di atas telah ditemukan oleh seorang hakim Perancis bernama Pierre de Fermat; dia menulis surat kepada para ahli lain seperti pendeta Marin Mersenne dan Blaise Pascal yang bekerja di bagian pajak penghasilan. Jadi titik unik F sedemikian rupa sehingga $FA+FB+FC$ minimal, disebut titik Fermat dari segitiga. Namun tampaknya beberapa tahun sebelumnya, Torricelli dari Italia telah menemukan titik ini sebelum Fermat menemukannya! Bagaimanapun juga, tradisinya adalah mencatat titik F ini. . .

Empat poin

Langkah selanjutnya adalah menambahkan titik ke-4 D dan mencoba meminimumkan $MA+MB+MC+MD$; misalkan Anda adalah operator TV kabel dan ingin menemukan di bidang mana Anda harus meletakkan antenna 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/EMT4Geometry_Shintia Nurchol

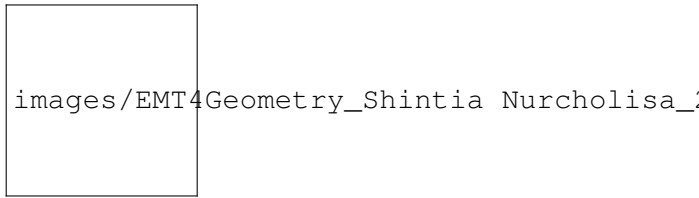
Gambar 12.325 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-139.png

```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
>P=(A_B_C_D)'; plot2d(P[1],P[2],points=1,add=1,color=12);
>insimg;
```

Masih ada nilai minimum dan tidak ada simpul A, B, C, maupun

D:

```
>function f(x):=d4(x[1],x[2])
>neldermin("f",[0.2,0.2])
```



Gambar 12.326 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-140.png

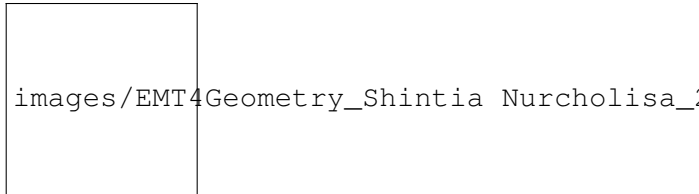
```
[0.142858, 0.142857]
```

Tampaknya dalam kasus ini, koordinat titik optimal adalah rasional atau mendekati rasional...

Karena ABCD adalah sebuah bujur sangkar, maka kita berharap bahwa titik optimalnya adalah pusat dari ABCD:

```
>C=[-1,1];
```

```
>plot3d("d4",xmin=-1,xmax=1,ymin=-1,ymax=1):
```



Gambar 12.327 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-141.png

```
>fcontour("d4",xmin=-1.5,xmax=1.5,ymin=-1.5,ymax=1.5,hue=1);
```

```
>P=(A.B.C.D)'; plot2d(P[1],P[2],add=1,color=12,points=1);
```

```
>insimg;
```



Gambar 12.328 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-142.png

CONTOH 7: BOLA DANDELIN DENGAN POV-Ray

Anda dapat menjalankan demonstrasi ini, jika Anda telah menginstal Povray, dan pvengine.exe pada path program.

Pertama, kita menghitung jari-jari bola.

Jika Anda melihat gambar di bawah ini, Anda dapat melihat bahwa kita membutuhkan dua lingkaran yang menyentuh dua garis yang membentuk kerucut, dan satu garis yang membentuk bidang yang memotong kerucut.

Kita menggunakan file geometri.e dari Euler untuk hal ini.

```
>load geometry;
```

Pertama, dua garis membentuk kerucut.

```
>g1 &= lineThrough([0,0],[1,a])
```

$$[-a, 1, 0]$$

```
>g2 &= lineThrough([0,0],[-1,a])
```

$$[-a, -1, 0]$$

Kemudian baris ketiga.

```
>g &= lineThrough([-1,0],[1,1])
```

$$[-1, 2, 1]$$

Kami merencanakan semuanya sejauh ini.

```
>setPlotRange(-1,1,0,2);
```

```
>color(black); plotLine(g(),"")
```

```
>a:=2; color(blue); plotLine(g1(),""), plotLine(g2(),""):
```

Sekarang, kita ambil titik umum pada sumbu y.

```
>P &= [0,u]
```

$$[0, u]$$



Gambar 12.329 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-143.png

Hitung jarak ke g1.

```
>d1 &= distance(P,projectToLine(P,g1)); $d1
```

$$\sqrt{\left(\frac{a^2 u}{a^2 + 1} - u\right)^2 + \frac{a^2 u^2}{(a^2 + 1)^2}}$$

Hitung jarak ke g.

```
>d &= distance(P,projectToLine(P,g)); $d
```

$$\sqrt{\left(\frac{u + 2}{5} - u\right)^2 + \frac{(2u - 1)^2}{25}}$$

Dan temukan pusat kedua lingkaran, yang jaraknya sama.

```
>sol &= solve(d1^2=d2,u); $sol
```

$$\left[u = \frac{-\sqrt{5} \sqrt{a^2 + 1} + 2a^2 + 2}{4a^2 - 1}, u = \frac{\sqrt{5} \sqrt{a^2 + 1} + 2a^2 + 2}{4a^2 - 1} \right]$$

Ada dua solusi.

Kami mengevaluasi solusi simbolis, dan menemukan kedua pusat, dan kedua jarak.

```
>u := sol()
```

```
[0.333333, 1]
```

```
>dd := d()
```

```
[0.149071, 0.447214]
```

Plot lingkaran-lingkaran tersebut ke dalam gambar.

```
>color(red);  
>plotCircle(circleWithCenter([0,u[1]],dd[1]),““);  
>plotCircle(circleWithCenter([0,u[2]],dd[2]),““);  
>insimg;
```



images/EMT4Geometry_Shintia Nurchol

Gambar 12.330 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-147.png

Plot dengan Povray

Selanjutnya kita plot semuanya dengan Povray. Perhatikan bahwa Anda mengubah perintah apapun pada urutan perintah Povray berikut ini, dan jalankan kembali semua perintah dengan Shift-Return.

Pertama kita memuat fungsi povray.

```
>load povray;  
>defaultpovray=“C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe”
```

C:\Program Files\POV-Ray\v3.7\bin\pvengine.exe

Kami menyiapkan pemandangan dengan tepat.

```
>povstart(zoom=11,center=[0,0,0.5],height=10°,angle=140°);
```

Selanjutnya kita tulis kedua bola tersebut ke file Povray.

```
>writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));  
>writeln(povsphere([0,0,u[2]],dd[2],povlook(red)));
```

Dan kerucutnya, transparan.

```
>writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray,1)));
```

Kami menghasilkan bidang yang terbatas pada kerucut.

```
>gp=g();
```

```
>pc=povcone([0,0,0],0,[0,0,a],1,"");
>vp=[gp[1],0,gp[2]]; dp=gp[3];
>writeln(povplane(vp,dp,povlook(blue,0.5),pc));
```

Sekarang kita menghasilkan dua titik pada lingkaran, di mana bola menyentuh kerucut.

```
>function turnz(v) := return
>P1=projectToLine([0,u[1]],g1()); P1=turnz([P1[1],0,P1[2]]);
>writeln(povpoint(P1,povlook(yellow)));
>P2=projectToLine([0,u[2]],g1()); P2=turnz([P2[1],0,P2[2]]);
>writeln(povpoint(P2,povlook(yellow)));
```

Kemudian, kita menghasilkan dua titik di mana bola-bola tersebut menyentuh bidang. Ini adalah fokus elips.

```
>P3=projectToLine([0,u[1]],g()); P3=[P3[1],0,P3[2]];
>writeln(povpoint(P3,povlook(yellow)));
>P4=projectToLine([0,u[2]],g()); P4=[P4[1],0,P4[2]];
>writeln(povpoint(P4,povlook(yellow)));
```

Selanjutnya kita menghitung perpotongan P1P2 dengan bidang.

```
>t1=scalp(vp,P1)-dp; t2=scalp(vp,P2)-dp; P5=P1+t1/(t1-
t2)*(P2-P1);
```

```
>writeln(povpoint(P5,povlook(yellow)));
```

Kami menghubungkan titik-titik dengan segmen garis.

```
>writeln(povsegment(P1,P2,povlook(yellow)));
>writeln(povsegment(P5,P3,povlook(yellow)));
>writeln(povsegment(P5,P4,povlook(yellow)));
```

Sekarang, kita menghasilkan pita abu-abu, di mana bola-bola menyentuh kerucut.

```
>pcw=povcone([0,0,0],0,[0,0,a],1.01);
>pc1=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P1[3]+defaultpointsize/2]
>writeln(povintersection([pcw,pc1],povlook(gray)));
>pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+defaultpointsize/2]
>writeln(povintersection([pcw,pc2],povlook(gray)));
```

Mulai program Povray.

```
>povend();
```



Gambar 12.331 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-148.png

Untuk mendapatkan Anaglyph ini, kita harus memasukkan semuanya ke dalam fungsi scene. Fungsi ini akan digunakan dua kali nanti.

>function scene () ...

```
global a,u,dd,g,g1,defaultpointsize;
writeln(povsphere([0,0,u[1]],dd[1],povlook(red)));
writeln(povsphere([0,0,u[2]],dd[2],povlook(red)));
writeln(povcone([0,0,0],0,[0,0,a],1,povlook(lightgray)));
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);
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);
pcl=povcylinder([0,0,P1[3]-defaultpointsize/2],[0,0,P2[3]-defaultpointsize/2],1.01);
```

```
writeln(povintersection([pcw,pc1],povlook(gray)));
pc2=povcylinder([0,0,P2[3]-defaultpointsize/2],[0,0,P2[3]+
writeln(povintersection([pcw,pc2],povlook(gray)));
endfunction
```

Anda memerlukan kacamata merah/sian untuk mengapresiasi efek berikut ini.

```
>povanaglyph("scene",zoom=11,center=[0,0,0.5],height=10°,angle=140°);
```



Gambar 12.332 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-149.png

CONTOH 8: GEOMETRI BUMI

Dalam buku catatan ini, kita ingin melakukan beberapa komputasi bola. Fungsi-fungsi tersebut terdapat pada file “spherical.e” di dalam folder examples. Kita perlu memuat file tersebut terlebih dahulu.

```
>load “spherical.e”;
```

Untuk memasukkan posisi geografis, kita menggunakan vektor dengan dua koordinat dalam radian (utara dan timur, nilai negatif untuk selatan dan barat). Berikut ini adalah koordinat untuk Kampus FMIPA UNY.

```
>FMIPA=[rad(-7,-46.467),rad(110,23.05)]
```

```
[-0.13569, 1.92657]
```

Anda dapat mencetak posisi ini dengan sposprint (cetak posisi bola).

```
>sposprint(FMIPA) // posisi garis lintang dan garis bujur  
FMIPA UNY
```

```
S 7°46.467' E 110°23.050'
```

Mari kita tambahkan dua kota lagi, Solo dan Semarang.

```
>Solo=[rad(-7,-34.333),rad(110,49.683)]; Semarang=[rad(-6,-  
59.05),rad(110,24.533)];
```

```
>sposprint(Solo), sposprint(Semarang),
```

```
S 7°34.333' E 110°49.683'
```

```
S 6°59.050' E 110°24.533'
```

Pertama, kita menghitung vektor dari satu titik ke titik lainnya pada bola ideal. Vektor ini adalah [heading, jarak] dalam radian. Untuk menghitung jarak di bumi, kita kalikan dengan jari-jari bumi pada garis lintang 7° .

```
>br=svector(FMIPA,Solo); degprint(br[1]), br[2]*rearth(7°)-  
>km // perkiraan jarak FMIPA-Solo
```


65°20'26.60''
53.8945384608

Ini adalah perkiraan yang baik. Rutinitas berikut ini menggunakan perkiraan yang lebih baik lagi. Pada jarak yang pendek, hasilnya hampir sama.

```
>esdist(FMIPA,Semarang)->" km" // perkiraan jarak FMIPA-Semarang
```

Commands must be separated by semicolon or comma!

Found: // perkiraan jarak FMIPA-Semarang (character 32)

You can disable this in the Options menu.

Error in:

```
esdist(FMIPA,Semarang)->" km" // perkiraan jarak FMIPA-Semarang
```

Ada fungsi untuk judul, dengan mempertimbangkan bentuk elips bumi. Sekali lagi, kami mencetak dengan cara yang canggih.

```
>sdegprint(esdir(FMIPA,Solo))
```

65.34°

Sudut segitiga melebihi 180° pada bola.

```
>asum=sangle(Solo,FMIPA,Semarang)+sangle(FMIPA,Solo,Semarang)+sangle(Solo,FMIPA,Semarang)
degprint(asum)
```

180°0'10.77''

Ini bisa digunakan untuk menghitung luas segitiga. Catatan: Untuk segitiga kecil, cara ini tidak akurat karena kesalahan pengurangan dalam asum-pi.

```
>(asum-pi)*rearth(48°)^2->" km^2" // perkiraan luas segitiga FMIPA-Solo-Semarang
```

Commands must be separated by semicolon or comma!

Found: // perkiraan luas segitiga FMIPA-Solo-Semarang (character 32)

You can disable this in the Options menu.

Error in:

```
(asum-pi)*rearth(48°)^2->" km^2" // perkiraan luas
```

Ada sebuah fungsi untuk hal ini, yang menggunakan garis lintang rata-rata segitiga untuk menghitung radius bumi, dan menangani kesalahan pembulatan untuk segitiga yang sangat kecil.

```
>esarea(Solo,FMIPA,Semarang)->" km^2", //perkiraan yang sama dengan fungsi esarea()
```

```
2123.64310526 km^2
```

Kita juga dapat menambahkan vektor ke posisi. Sebuah vektor berisi arah dan jarak, keduanya dalam radian. Untuk mendapatkan sebuah vektor, kita menggunakan svector. Untuk menambahkan sebuah vektor ke sebuah posisi, kita menggunakan saddvector.

```
>v=svector(FMIPA,Solo); sposprint(saddvector(FMIPA,v)), sposprint(Solo),
```

```
S 7°34.333' E 110°49.683'  
S 7°34.333' E 110°49.683'
```

Fungsi-fungsi ini mengasumsikan bola yang ideal. Hal yang sama di bumi.

```
>sposprint(esadd(FMIPA,esdir(FMIPA,Solo),esdist(FMIPA,Solo))), sposprint(Solo),
```

```
S 7°34.333' E 110°49.683'  
S 7°34.333' E 110°49.683'
```

Mari kita beralih ke contoh yang lebih besar, Tugu Jogja dan Monas Jakarta (menggunakan Google Earth untuk menemukan koordinatnya).

```
>Tugu=[-7.7833°,110.3661°]; Monas=[-6.175°,106.811944°];  
>sposprint(Tugu), sposprint(Monas)
```

```
S 7°46.998' E 110°21.966'  
S 6°10.500' E 106°48.717'
```

Menurut Google Earth, jaraknya adalah 429,66 km. Kami mendapatkan perkiraan yang bagus.

`>esdist(Tugu,Monas)->" km" // perkiraan jarak Tugu Jogja - Monas Jakarta`

Commands must be separated by semicolon or comma!

Found: // perkiraan jarak Tugu Jogja - Monas Jakarta (cha

You can disable this in the Options menu.

Error in:

`esdist(Tugu,Monas)->" km" // perkiraan jarak Tugu Jogja`
^

Judulnya sama dengan yang dihitung di Google Earth.

`>degprint(esdir(Tugu,Monas))`

294°17'2.85''

Namun demikian, kita tidak lagi mendapatkan posisi target yang tepat, jika kita menambahkan arah dan jarak ke posisi semula. Hal ini terjadi, karena kita tidak menghitung fungsi inversi secara tepat, tetapi mengambil perkiraan radius bumi di sepanjang jalur.

`>sposprint(esadd(Tugu,esdir(Tugu,Monas),esdist(Tugu,Monas)))`

S 6°10.500' E 106°48.717'

Namun demikian, kesalahannya tidak besar.

`>sposprint(Monas),`

S 6°10.500' E 106°48.717'

Tentu saja, kita tidak bisa berlayar dengan arah yang sama dari satu tujuan ke tujuan lainnya, jika kita ingin mengambil jalur terpendek. Bayangkan, Anda terbang ke arah NE mulai dari titik mana pun di bumi. Kemudian Anda akan berputar ke kutub utara. Lingkaran besar tidak mengikuti arah yang konstan!

Perhitungan berikut ini menunjukkan bahwa kita akan melenceng dari tujuan yang benar, jika kita menggunakan arah yang sama selama perjalanan.

```
>dist=esdist(Tugu,Monas); hd=esdir(Tugu,Monas);
```

Sekarang kita tambahkan 10 kali sepersepuluh dari jarak tersebut, dengan menggunakan arah menuju Monas, kita akan sampai di Tugu.

```
>p=Tugu; loop 1 to 10; p=esadd(p,hd,dist/10); end;
```

Hasilnya jauh berbeda.

```
>sposprint(p), skmprint(esdist(p,Monas))
```

S 6°11.250' E 106°48.372'
1.529km

Sebagai contoh lain, mari kita ambil dua titik di bumi pada garis lintang yang sama.

```
>P1=[30°,10°]; P2=[30°,50°];
```

Jalur terpendek dari P1 ke P2 bukanlah lingkaran lintang 30°, tetapi jalur yang lebih pendek yang dimulai 10° lebih jauh ke utara di P1.

```
>sdegprint(esdir(P1,P2))
```

79.69°

Namun, jika kita mengikuti pembacaan kompas ini, kita akan berputar ke kutub utara! Jadi, kita harus menyesuaikan arah kita di sepanjang jalan. Untuk tujuan kasar, kita sesuaikan pada 1/10 dari jarak total.

```
>p=P1; dist=esdist(P1,P2); ...
```

```
> loop 1 to 10; dir=esdir(p,P2); sdegprint(dir), p=esadd(p,dir,dist/10);  
end;
```

79.69°

81.67°

83.71°

85.78°

87.89°

90.00°

92.12°

```
94.22°  
96.29°  
98.33°
```

Jaraknya tidak tepat, karena kita akan menambahkan sedikit kesalahan, jika kita mengikuti judul yang sama terlalu lama.

```
>skmprint(esdist(p,P2))
```

```
0.203km
```

Kita akan mendapatkan perkiraan yang baik, jika kita menyesuaikan arah setiap 1/100 dari total jarak dari Tugu ke Monas.

```
>p=Tugu; dist=esdist(Tugu,Monas); ...  
> loop 1 to 100; p=esadd(p,esdir(p,Monas),dist/100); end;  
>skmprint(esdist(p,Monas))
```

```
0.000km
```

Untuk keperluan navigasi, kita bisa mendapatkan urutan posisi GPS di sepanjang Bundaran Hotel Indonesia menuju Monas dengan fungsi `navigate`.

```
>load spherical; v=navigate(Tugu,Monas,10); ...  
> loop 1 to rows(v); sposprint(v[#]), end;
```

```
S 7°46.998' E 110°21.966'  
S 7°37.422' E 110°0.573'  
S 7°27.829' E 109°39.196'  
S 7°18.219' E 109°17.834'  
S 7°8.592' E 108°56.488'  
S 6°58.948' E 108°35.157'  
S 6°49.289' E 108°13.841'  
S 6°39.614' E 107°52.539'  
S 6°29.924' E 107°31.251'  
S 6°20.219' E 107°9.977'  
S 6°10.500' E 106°48.717'
```

Kami menulis sebuah fungsi, yang memplot bumi, dua posisi, dan posisi di antaranya.

```
>function testplot ...
```

```

useglobal;
plotearth;
plotpos(Tugu,"Tugu Jogja"); plotpos(Monas,"Tugu Monas");
plotposline(v);
endfunction

```

Sekarang rencanakan semuanya.

```
>plot3d("testplot",angle=25, height=6,>own,>user,zoom=4):
```



Gambar 12.333 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-150.png

Atau gunakan plot3d untuk mendapatkan tampilan anaglyph. Ini terlihat sangat bagus dengan kacamata merah/cyan.

```
>plot3d("testplot",angle=25,height=6,distance=5,own=1,anaglyph=1,zoom=4):
```



Gambar 12.334 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-151.png

LATIHAN

1. Gambarkanlah segi- n beraturan jika diketahui titik pusat O , n , dan jarak titik pusat ke titik-titik sudut segi- n tersebut (jari-jari lingkaran luar segi- n), r .

Petunjuk:

- Besar sudut pusat yang menghadap masing-masing sisi segi- n adalah $(360/n)$.
- Titik-titik sudut segi- n merupakan perpotongan lingkaran luar segi- n dan garis-garis yang melalui pusat dan saling membentuk sudut sebesar kelipatan $(360/n)$.
- Untuk n ganjil, pilih salah satu titik sudut adalah di atas.
- Untuk n genap, pilih 2 titik di kanan dan kiri lurus dengan titik pusat.
- Anda dapat menggambar segi-3, 4, 5, 6, 7, dst beraturan.

>load geometry

Numerical and symbolic geometry.

>setPlotRange(-3.5,3.5,-3.5,3.5):



images/EMT4Geometry_Shintia Nurchol

Gambar 12.335 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-152.png



Gambar 12.336 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-153.png



Gambar 12.337 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-154.png

```
>A=[-2,-2]; plotPoint(A,"A");  
>B=[2,-2]; plotPoint(B,"B");  
>C=[0,3]; plotPoint(C,"C");
```



Gambar 12.338 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-155.png

```
>plotSegment(A,B,"a");  
>plotSegment(B,C,"b");  
>plotSegment(A,C,"c");  
>c=circleThrough(A,B,C);  
>R=getCircleRadius(c);
```



Gambar 12.339 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-156.png

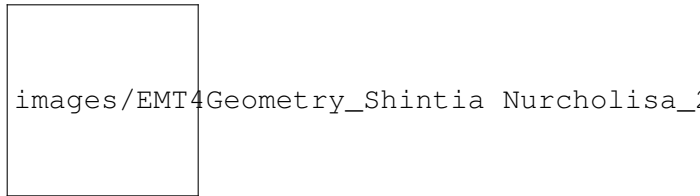


Gambar 12.340 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-157.png



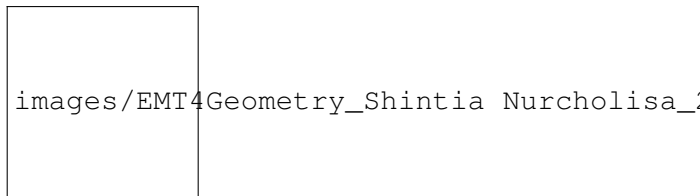
Gambar 12.341 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-158.png

```
>O=getCircleCenter(c);
>plotPoint(O,“O”):
```



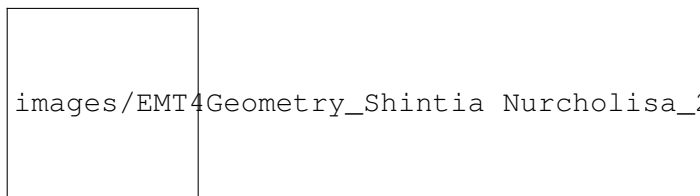
Gambar 12.342 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-159.png

```
>l=angleBisector(A,B,C);
>color(2); plotLine(l); color(1):
```



Gambar 12.343 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-160.png

```
>plotCircle(c,“Lingkaran luar segitiga ABC”):
```



Gambar 12.344 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-161.png

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 .

```
>load geometry;
>setPlotRange(5); P=[1,0]; Q=[4,0]; R=[0,-4];
>plotPoint(P,"P"); plotPoint(Q,"Q"); plotPoint(R,"R");
```



Gambar 12.345 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-162.png

```
>sol &= solve([a+b=-c,16*a+4*b=-c,c=-4],[a,b,c])

[[a = - 1, b = 5, c = - 4]]

>function y&=-x^2+5*x-4

      2
    - x  + 5 x - 4

>plot2d("-x^2+5*x-4",-5,5,-5,5):
```

3. Gambarkanlah suatu segi-4 yang diketahui keempat titik sudutnya, misalnya A, B, C, D.
 - Tentukan apakah segi-4 tersebut merupakan segi-4 garis singgung (sisinya-sisintya merupakan garis singgung lingkaran yang sama yakni lingkaran dalam segi-4 tersebut).



Gambar 12.346 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-163.png

- Suatu segi-4 merupakan segi-4 garis singgung apabila keempat garis bagi sudutnya bertemu di satu titik.
- Jika segi-4 tersebut merupakan segi-4 garis singgung, gambar lingkaran dalamnya.
- Tunjukkan bahwa syarat suatu segi-4 merupakan segi-4 garis singgung apabila hasil kali panjang sisi-sisi yang berhadapan sama.

```
>load geometry
```

Numerical and symbolic geometry.

```
>setPlotRange(-3.5,3.5,-3.5,3.5):
```



Gambar 12.347 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-164.png

```
>A=[-2,-2]; plotPoint(A,"A");  
>B=[2,-2]; plotPoint(B,"B");  
>C=[2,2]; plotPoint(C,"C");  
>D=[-2,2]; plotPoint(D,"D");
```



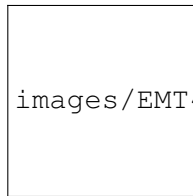
Gambar 12.348 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-165.png



Gambar 12.349 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-166.png



Gambar 12.350 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-167.png



Gambar 12.351 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-168.png

```

>plotSegment(A,B,“”);
>plotSegment(B,C,“”);
>plotSegment(C,D,“”);
>plotSegment(A,D,“”);
>aspect(1):

```



Gambar 12.352 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-169.png

```

>l=angleBisector(A,B,C);
>m=angleBisector(B,C,D);
>P=lineIntersection(l,m);
>color(5); plotLine(l); plotLine(m); color(1);
>plotPoint(P,“P”):

```



Gambar 12.353 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-170.png

```

>r=norm(P-projectToLine(P,lineThrough(A,B)));
>plotCircle(circleWithCenter(P,r,“Lingkaran dalam segiempat
ABCD”):
>AB=norm(A-B) //panjang sisi AB

```



images/EMT4Geometry_Shintia Nurchol

Gambar 12.354 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-171.png

```
>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

```
>AD.BC
```

16

Terlihat bahwa,

$$AB \times CD = 16, AD \times BC = 16$$

sehingga terbukti bahwa

$$AB \times CD, AD \times BC$$

karena semua syarat telah terpenuhi, maka dapat dipastikan bahwa segiempat tersebut merupakan segiempat garis singgung.

4. Gambarlah suatu ellips jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang jumlah jarak ke P dan ke Q selalu sama (konstan).

```
>P=[-1,-1]; Q=[1,-1];
>function d1(x,y):=sqrt((x-P[1])2+(y-P[2])2)
>Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])2+(y-P[2])2)+sqrt((x-
Q[1])2+(y-Q[2])2)
>fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
```



Gambar 12.355 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-174.png

```
>plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
```

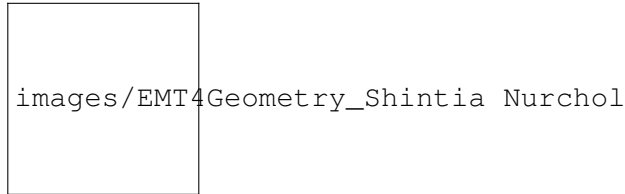


Gambar 12.356 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-175.png

5. Gambarlah suatu hiperbola jika diketahui kedua titik fokusnya, misalnya P dan Q. Ingat ellips dengan fokus P dan Q adalah tempat kedudukan titik-titik yang selisih jarak ke P dan ke Q selalu sama (konstan).

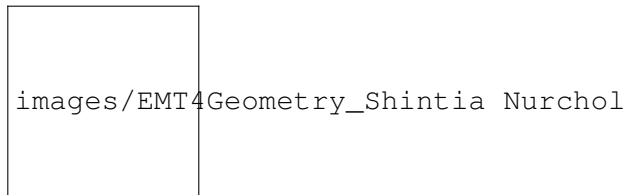
```
>P=[-1,-1]; Q=[1,-1];
```

```
>function d1(x,y):=sqrt((x-p[1])2+(y-p[2])2)
>Q=[1,-1]; function d2(x,y):=sqrt((x-P[1])2+(y-P[2])2)+sqrt((x+Q[1])2+(y+Q[2])2)
>fcontour("d2",xmin=-2,xmax=2,ymin=-3,ymax=1,hue=1):
```



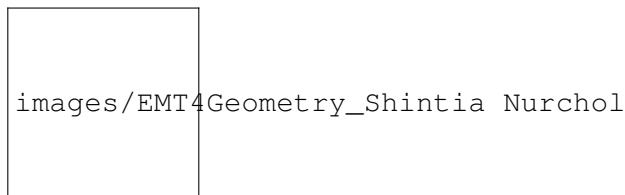
Gambar 12.357 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-176.png

```
>plot3d("d2",xmin=-2,xmax=2,ymin=-3,ymax=1):
```



Gambar 12.358 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-177.png

```
>plot2d("abs(x+1)+abs(x-1)",xmin=-3,xmax=3):
```



Gambar 12.359 images/EMT4Geometry_Shintia%20Nurcholisa_23030630071-178.png

unicodehyperref hyphensurl []book xcolor amsmath,amssymb
iftex [T1]fontenc [utf8]inputenc textcomp lmodern upquote []microtype
[protrusion]basicmath parskip graphicx bookmark xurl

EMT UNTUK STATISTIKA

Di buku catatan ini, kami mendemonstrasikan plot statistik utama, pengujian, dan distribusi di Euler.

Mari kita mulai dengan beberapa statistik deskriptif. Ini bukan pengantar statistik. Jadi, Anda mungkin memerlukan latar belakang untuk memahami detailnya.

Asumsikan pengukuran berikut. Kami ingin menghitung nilai rata-rata dan deviasi standar yang diukur.

```
>M=[1000,1004,998,997,1002,1001,998,1004,998,997]; ...  
> median(M), mean(M), dev(M),
```

```
999  
999.9  
2.72641400622
```

Kita dapat memplot plot kotak-dan-kumis untuk datanya. Dalam kasus kami, tidak ada outlier.

```
>aspect(1.75); boxplot(M):
```



Gambar 12.360 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-001.png

Kami menghitung probabilitas suatu nilai lebih besar dari 1005, dengan asumsi nilai terukur berdistribusi normal.

Semua fungsi untuk distribusi di Euler diakhiri dengan ...dis dan menghitung distribusi probabilitas kumulatif (CPF).

$$\text{normaldis}(x,m,d) = \int_{-\infty}^x \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-m}{d}\right)^2} dt.$$

Kami mencetak hasilnya dalam % dengan akurasi 2 digit menggunakan fungsi 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 tertentu.

```
>r=155.5:4:187.5; v=[22,71,136,169,139,71,32,8];
```

Berikut adalah alur pendistribusiannya.

```
>plot2d(r,v,a=150,b=200,c=0,d=190,bar=1,style="\"):
```



images/EMT4Statistika_Shintia Nurcholisa

Gambar 12.361 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-003.png

Kita bisa memasukkan data mentah tersebut ke dalam tabel.

Tabel adalah metode untuk menyimpan data statistik. Tabel kita harus berisi tiga kolom: Awal jangkauan, akhir jangkauan, jumlah pria dalam jangkauan.

Tabel dapat dicetak dengan header. Kami menggunakan vektor string untuk mengatur header.

```
>T:=r[1:8]' | r[2:9]' | v'; writetable(T,labc=["BB","BA","Frek"])
```

BB	BA	Frek
155.5	159.5	22
159.5	163.5	71
163.5	167.5	136
167.5	171.5	169
171.5	175.5	139
175.5	179.5	71

179.5	183.5	32
183.5	187.5	8

Jika kita memerlukan nilai rata-rata dan statistik ukuran lainnya, kita perlu menghitung titik tengah rentang tersebut. Kita bisa menggunakan dua kolom pertama tabel kita untuk ini.

Symbol “|” digunakan untuk memisahkan kolom, fungsi “writetable” digunakan untuk menulis tabel, dengan opsi “labc” 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 akan lebih mudah jika menjumlahkan 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]
```

Sekarang kita dapat menghitung mean dan deviasi sampel dengan frekuensi tertentu.

```
>{m,d}=meandev(M,v); m, d,
```

```
169.901234568
5.98912964449
```

Mari kita tambahkan distribusi nilai normal ke diagram batang di atas. Rumus distribusi normal dengan mean m dan simpangan baku d adalah:

$$y = \frac{1}{d\sqrt{2\pi}} e^{\frac{-(x-m)^2}{2d^2}}.$$

Karena nilainya antara 0 dan 1, maka untuk memplotnya pada bar plot harus dikalikan dengan 4 kali jumlah data.

```
> plot2d("qnormal(x,m,d)*sum(v)*4", ...  
> xmin=min(r),xmax=max(r),thickness=3,add=1):
```



images/EMT4Statistika_Shintia Nurcholisa

Gambar 12.362 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-005.png

TABEL

Di direktori buku catatan ini Anda menemukan file dengan tabel. Data tersebut merupakan hasil survei. Berikut adalah empat baris pertama file tersebut. Datanya berasal dari buku online Jerman “Einführung in die Statistik mit R” oleh A. Handl.

```
>printfile (“table.dat”, 7);
```

```
Could not open the file  
table.dat  
for reading!
```

```
Try "trace errors" to inspect local variables after e  
printfile:
```

```
open(filename, "r");
```

Tabel berisi 7 kolom angka atau token (string). Kami ingin membaca tabel dari file. Pertama, kami menggunakan terjemahan kami sendiri untuk tokennya.

Untuk ini, kami mendefinisikan kumpulan token. Fungsi `strtokens()` mendapatkan vektor string token dari string tertentu.

```
>mf=[“m”,“f”]; yn=[“y”,“n”]; ev:=strtokens(“g vg m b vb”);
```

Sekarang kita membaca tabel dengan terjemahan ini.

Argumen `tok2`, `tok4` dll. adalah terjemahan dari kolom tabel. Argumen ini tidak ada dalam daftar parameter `readtable()`, jadi Anda perlu menyediakannya dengan “:=”.

```
>{MT,hd}=readtable(“table.dat”,tok2:=mf,tok4:=yn,tok5:=ev,tok7:=yn);
```

```
Could not open the file  
table.dat  
for reading!
```

```
Try "trace errors" to inspect local variables after e  
readtable:
```

```
if filename!=none then open(filename, "r"); endif;
```

```
>load over statistics;
```

Untuk mencetak, kita perlu menentukan kumpulan 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);
```

```
MT is not a variable!
```

```
Error in:
```

```
writetable(MT[1:10],labc=hd,wc=5,tok2:=mf,tok4:=yn,tok5:=e
^
```

Titik “.” mewakili nilai-nilai, yang tidak tersedia.

Jika kita tidak ingin menentukan token yang akan diterjemahkan terlebih dahulu, kita hanya perlu menentukan, kolom mana yang berisi token dan bukan angka.

```
>ctok=[2,4,5,7]; {MT,hd,tok}=readtable("table.dat",ctok=ctok);
```

```
Could not open the file
```

```
table.dat
```

```
for reading!
```

```
Try "trace errors" to inspect local variables after errors
```

```
readtable:
```

```
if filename!=none then open(filename,"r"); endif;
```

Fungsi readtable() kini mengembalikan sekumpulan token.

```
>tok
```

```
Variable tok not found!
```

```
Error in:
```

```
tok ...
```

```
^
```

Tabel berisi entri dari file dengan token yang diterjemahkan ke dalam angka.

String khusus NA = “.” diartikan sebagai “Tidak Tersedia”, dan mendapatkan NAN (bukan angka) di tabel. Terjemahan ini dapat diubah dengan parameter NA, dan NAval.

```
>MT[1]
```

```
MT is not a variable!
Error in:
MT[1] ...
^
```

Berikut isi tabel dengan nomor yang belum diterjemahkan.

```
>writetable(MT,wc=5)
```

```
Variable or function MT not found.
Error in:
writetable(MT,wc=5) ...
^
```

Untuk kenyamanan, Anda dapat memasukkan keluaran `readtable()` ke dalam daftar.

```
>Table={{readtable("table.dat",ctok=ctok)}};
```

```
Could not open the file
table.dat
for reading!
Try "trace errors" to inspect local variables after e
readtable:
  if filename!=none then open(filename,"r"); endif;
```

Dengan menggunakan kolom token yang sama dan token yang dibaca dari file, kita dapat mencetak tabel. Kita dapat menentukan `ctok`, `tok`, dll. atau menggunakan tabel daftar.

```
>writetable(Table,ctok=ctok,wc=5);
```

```
Variable or function Table not found.
Error in:
writetable(Table,ctok=ctok,wc=5); ...
^
```

Fungsi `tablecol()` mengembalikan nilai kolom tabel, melewati baris apa pun dengan nilai NAN (“.” dalam file), dan indeks kolom, yang berisi nilai-nilai ini.

```
>{c,i}=tablecol(MT,[5,6]);
```

```
Variable or function MT not found.
Error in:
{c,i}=tablecol(MT,[5,6]); ...
      ^
```

Kita bisa menggunakan ini untuk mengekstrak kolom dari tabel untuk tabel baru.

```
>j=[1,5,6]; writetable(MT[i,j],lab=hd[j],ctok=[2],tok=tok)
```

```
Variable or function i not found.
Error in:
j=[1,5,6]; writetable(MT[i,j],lab=hd[j],ctok=[2],tok=tok)
      ^
```

Tentu saja, kita perlu mengekstrak tabel itu sendiri dari daftar Tabel dalam kasus ini.

```
>MT=Table[1];
```

```
Table is not a variable!
Error in:
MT=Table[1]; ...
      ^
```

Tentu saja, kita juga dapat menggunakannya untuk menentukan nilai rata-rata suatu kolom atau nilai statistik lainnya.

```
>mean(tablecol(MT,6))
```

```
Variable or function MT not found.
Error in:
mean(tablecol(MT,6)) ...
      ^
```

Fungsi `getstatistics()` mengembalikan elemen dalam vektor, dan jumlahnya. Kami menerapkannya pada nilai “m” dan “f” di kolom kedua tabel kami.

```
>{xu,count}=getstatistics(tablecol(MT,2)); xu, count,
```

```
Variable or function MT not found.
Error in:
{xu,count}=getstatistics(tablecol(MT,2)); xu, count,
^
```

Kita bisa mencetak hasilnya di tabel baru.

```
>writetable(count',labr=tok[xu])
```

```
Variable count not found!
Error in:
writetable(count',labr=tok[xu]) ...
^
```

Fungsi selecttable() mengembalikan tabel baru dengan nilai dalam satu kolom yang dipilih dari vektor indeks. Pertama kita mencari indeks dari dua nilai kita di tabel token.

```
>v:=indexof(tok,["g","vg"])
```

```
Variable or function tok not found.
Error in:
v:=indexof(tok,["g","vg"]) ...
^
```

Sekarang kita dapat memilih baris tabel, yang memiliki salah satu nilai v pada baris ke-5.

```
>MT1:=MT[selectrows(MT,5,v)]; i:=sortedrows(MT1,5);
```

```
Variable or function MT not found.
Error in:
MT1:=MT[selectrows(MT,5,v)]; i:=sortedrows(MT1,5); ..
^
```

Sekarang kita dapat mencetak tabel, dengan nilai yang diekstraksi dan diurutkan di kolom ke-5.

```
>writetable(MT1[i],labc=hd,ctok=ctok,tok=tok,wc=7);
```

```
Variable or function i not found.
```

Error in:

```
writetable(MT1[i], labc=hd, ctok=ctok, tok=tok, wc=7); ...  
^
```

Untuk statistik selanjutnya, kami ingin menghubungkan dua kolom tabel. Jadi kita ekstrak kolom 2 dan 4 dan urutkan tabelnya.

```
>i=sortedrows(MT,[2,4]); ...  
> writetable(tablecol(MT[i],[2,4]),ctok=[1,2],tok=tok)
```

Variable or function MT not found.

Error in:

```
i=sortedrows(MT,[2,4]); writetable(tablecol(MT[i],[2,4])  
^
```

Dengan `getstatistics()`, kita juga bisa menghubungkan jumlah dalam dua kolom tabel satu sama lain.

```
>MT24=tablecol(MT,[2,4]); ...  
> {xu1,xu2,count}=getstatistics(MT24[1],MT24[2]); ...  
> writetable(count,labr=tok[xu1],labc=tok[xu2])
```

Variable or function MT not found.

Error in:

```
MT24=tablecol(MT,[2,4]); {xu1,xu2,count}=getstatistics(MT24  
^
```

Sebuah tabel dapat ditulis ke file.

```
>filename="test.dat"; ...  
> writetable(count,labr=tok[xu1],labc=tok[xu2],file=filename);
```

Variable or function count not found.

Error in:

```
filename="test.dat"; writetable(count,labr=tok[xu1],labc=t  
^
```

Kemudian kita bisa membaca tabel dari file tersebut.

```
>{MT2,hd,tok2,hdr}=readtable(filename,>clabs,>rlabs); ...  
> writetable(MT2,labr=hdr,labc=hd)
```

Could not open the file
test.dat
for reading!

Try "trace errors" to inspect local variables after error.
readtable:

```
if filename~=none then open(filename,"r"); endif;
```

Dan hapus file tersebut.

```
>fileremove(filename);
```


DISTRIBUSI

Dengan plot2d, ada metode yang sangat mudah untuk memplot sebaran data eksperimen.

```
>p=normal(1,1000); //1000 random normal-distributed sample  
p  
>plot2d(p,distribution=20,style="\f"); // plot the random  
sample p  
>plot2d("qnormal(x,0,1)",add=1): // add the standard normal  
distribution plot
```



Gambar 12.363 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-006.png

Perlu diperhatikan perbedaan antara bar plot (sampel) dan kurva normal (distribusi sebenarnya). Masukkan kembali ketiga perintah untuk melihat hasil pengambilan sampel lainnya.

Berikut adalah perbandingan 10 simulasi dari 1000 nilai terdistribusi normal menggunakan apa yang disebut plot kotak. Plot ini menunjukkan median, kuartil 25% dan 75%, nilai minimal dan maksimal, serta outlier.

```
>p=normal(10,1000); boxplot(p):
```

Untuk menghasilkan bilangan bulat acak, Euler memiliki intrandom. Mari kita simulasikan lemparan dadu dan plot distribusinya.

Kita menggunakan fungsi getmultiplicities(v,x), yang menghitung seberapa sering elemen v muncul di x. Kemudian kita plot hasilnya menggunakan kolomplot().

```
>k=intrandom(1,6000,6); ...
```



images/EMT4Statistika_Shintia Nurcholis

Gambar 12.364 images/EMT4Statistika_Shintia%20Nurcholis_23030630071-007.png

```
> columnsplot(getmultiplicities(1:6,k)); ...
> ygrid(1000,color=red):
```



images/EMT4Statistika_Shintia Nurcholis

Gambar 12.365 images/EMT4Statistika_Shintia%20Nurcholis_23030630071-008.png

Meskipun `inrandom(n,m,k)` mengembalikan bilangan bulat yang terdistribusi secara seragam dari 1 hingga k, distribusi bilangan bulat lainnya dapat digunakan dengan `randpint()`.

Dalam contoh berikut, probabilitas untuk 1,2,3 masing-masing adalah 0,4,0.1,0.5.

```
>randpint(1,1000,[0.4,0.1,0.5]); getmultiplicities(1:3,%)
```

```
[378, 102, 520]
```

Euler dapat menghasilkan nilai acak dari lebih banyak distribusi. Lihat referensinya.

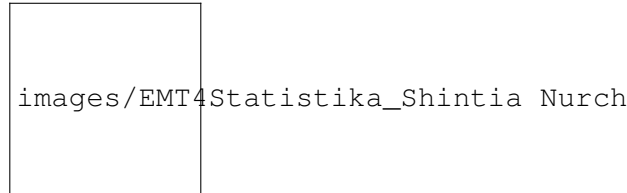
Misalnya, kita mencoba distribusi eksponensial. Variabel acak kontinu X dikatakan berdistribusi eksponensial, jika PDF-nya diberikan oleh

$$f_X(x) = \lambda e^{-\lambda x}, \quad x > 0, \quad \lambda > 0,$$

dengan parameter

$\lambda = \frac{1}{\mu}$, μ adalah mean, dan dilambangkan dengan $X \sim \text{Exponential}(\lambda)$.

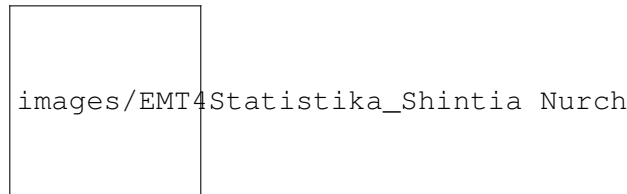
```
>plot2d(randexponential(1,1000,2),>distribution):
```



Gambar 12.366 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-011.png

Untuk banyak distribusi, Euler dapat menghitung fungsi distribusi dan inversnya.

```
>plot2d("normaldis",-4,4):
```



Gambar 12.367 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-012.png

Berikut ini adalah salah satu cara untuk memplot kuantil.

```
>plot2d("qnormal(x,1,1.5)",-4,6); ...
> plot2d("qnormal(x,1,1.5)",a=2,b=5,>add,>filled):
```

$$\text{normaldis}(x,m,d) = \int_{-\infty}^x \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t-m}{d})^2} dt.$$

Peluang berada di kawasan hijau adalah sebagai berikut.

```
>normaldis(5,1,1.5)-normaldis(2,1,1.5)
```



images/EMT4Statistika_Shintia Nurcholis

Gambar 12.368 images/EMT4Statistika_Shintia%20Nurcholis_23030630071-013.png

0.248662156979

Ini dapat dihitung secara numerik dengan 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 yang mean dan deviasinya sama. Fungsi `invbindis()` menyelesaikan interpolasi linier antara nilai integer.

```
>invbindis(0.95,1000,0.5), invnormaldis(0.95,500,0.5*sqrt(1000))
```

525.516721219

526.007419394

Fungsi `qdis()` adalah kepadatan distribusi chi-kuadrat. Seperti biasa, Euler memetakan vektor ke fungsi ini. Dengan demikian kita mendapatkan plot semua distribusi chi-kuadrat dengan derajat 5 sampai 30 dengan mudah dengan cara berikut.

```
>plot2d("qchidis(x,(5:5:50))",0,50):
```

Euler memiliki fungsi akurat untuk mengevaluasi distribusi. Mari kita periksa `chidis()` dengan integral.

Penamaannya mencoba untuk konsisten. Misalnya.,

- distribusi chi-kuadratnya adalah `chidis()`,



Gambar 12.369 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-016.png

- fungsi kebalikannya adalah `invchidis()`,
- kepadatannya adalah `qchidis()`.

Pelengkap distribusi (ekor atas) adalah `chicdis()`.

`>chidis(1.5,2), integrate("qchidis(x,2)",0,1.5)`

0.527633447259

0.527633447259

DISTRIBUSI DISKRIT

Untuk menentukan distribusi diskrit Anda sendiri, Anda dapat menggunakan metode berikut.

Pertama kita atur fungsi distribusinya.

```
>wd = 0|((1:6)+[-0.01,0.01,0,0,0,0])/6
```

```
[0, 0.165, 0.335, 0.5, 0.666667, 0.833333, 1]
```

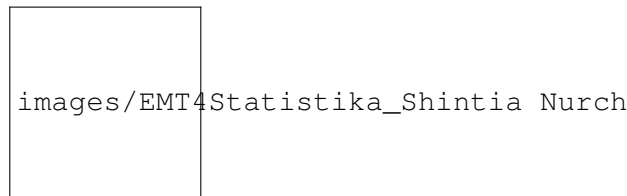
Artinya dengan probabilitas $wd[i+1]-wd[i]$ kita menghasilkan nilai acak i .

Ini hampir merupakan distribusi yang seragam. Mari kita tentukan generator nomor acak untuk ini. Fungsi $find(v,x)$ mencari nilai x pada vektor v . Fungsi ini juga berfungsi untuk vektor x .

```
>function wrongdice (n,m) := find(wd,random(n,m))
```

Kesalahannya sangat halus sehingga kita hanya melihatnya dengan banyak iterasi.

```
>columnsplot(getmultiplicities(1:6,wrongdice(1,1000000))):
```



Gambar 12.370 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-017.png

Berikut adalah fungsi sederhana untuk memeriksa keseragaman distribusi nilai $1 \dots K$ 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
```

Memang fungsinya 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 `binomialsun()`, yang mengembalikan probabilitas i atau kurang hit dari n percobaan.

```
>bindis(410,1000,0.4)
```

0.751401349654

Fungsi Beta terbalik digunakan untuk menghitung interval kepercayaan Clopper-Pearson untuk parameter p . Tingkat defaultnya adalah alfa.

Arti dari interval ini adalah jika p berada di luar interval, hasil pengamatan 410 dalam 1000 jarang terjadi.

```
>clopperpearson(410,1000)
```

[0.37932, 0.441212]

Perintah berikut adalah cara langsung untuk mendapatkan hasil di atas. Namun untuk n yang besar, penjumlahan langsungnya tidak akurat dan lambat.

```
>p=0.4; i=0:410; n=1000; sum(bin(n,i)*pi*(1-p)(n-i))
```

0.751401349655

```
invbinsum() menghitung kebalikan dari binomialsum().
>invbindis(0.75,1000,0.4)
```

```
409.932733047
```

Di Bridge, kami mengasumsikan 5 kartu beredar (dari 52) di dua tangan (26 kartu). Mari kita hitung probabilitas distribusi yang lebih buruk dari 3:2 (misalnya 0:5, 1:4, 4:1, atau 5:0).

```
>2*hypergeomsum(1,5,13,26)
```

```
0.321739130435
```

Ada juga simulasi distribusi multinomial.

```
>randmultinomial(10,1000,[0.4,0.1,0.5])
```

381	100	519
376	91	533
417	80	503
440	94	466
406	112	482
408	94	498
395	107	498
399	96	505
428	87	485
400	99	501

MERENCANAKAN DATA

Untuk memetakan data, kami mencoba hasil pemilu Jerman sejak tahun 1990, diukur dalam jumlah kursi.

```
>BW := [ ...  
> 1990,662,319,239,79,8,17; ...  
> 1994,672,294,252,47,49,30; ...  
> 1998,669,245,298,43,47,36; ...  
> 2002,603,248,251,47,55,2; ...  
> 2005,614,226,222,61,51,54; ...  
> 2009,622,239,146,93,68,76; ...  
> 2013,631,311,193,0,63,64];
```

Untuk pesta, kami menggunakan rangkaian nama.

```
>P:=[“CDU/CSU”,“SPD”,“FDP”,“Gr”,“Li”];
```

Mari kita cetak persentasenya dengan baik.

Pertama kita mengekstrak kolom yang diperlukan. Kolom 3 sampai 7 adalah kursi masing-masing partai, dan kolom 2 adalah jumlah kursi seluruhnya. kolom adalah tahun pemilihan.

```
>BT:=BW[,3:7]; BT:=BT/sum(BT); YT:=BW[,1]’;
```

Kemudian statistiknya kita cetak dalam bentuk tabel. Kami menggunakan nama sebagai header kolom, dan tahun sebagai header untuk baris. Lebar default untuk kolom adalah $wc=10$, tetapi kami lebih memilih keluaran yang lebih padat. Kolom akan diperluas untuk label kolom, jika perlu.

```
>writetable(BT*100,wc=6,dc=0,>fixed,labc=P,labr=YT)
```

	CDU/CSU	SPD	FDP	Gr	Li
1990	48	36	12	1	3
1994	44	38	7	7	4
1998	37	45	6	7	5
2002	41	42	8	9	0
2005	37	36	10	8	9
2009	38	23	15	11	12
2013	49	31	0	10	10

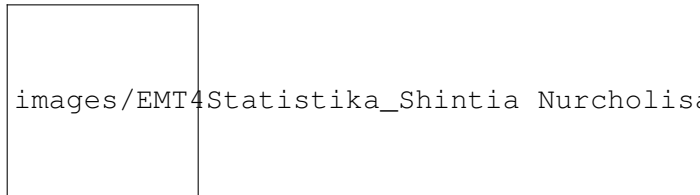
Perkalian matriks berikut ini menjumlahkan persentase dua partai besar yang menunjukkan bahwa partai-partai kecil berhasil memperoleh suara di parlemen hingga tahun 2009.

```
>BT1:=(BT.[1;1;0;0;0])'*100
```

```
[84.29, 81.25, 81.1659, 82.7529, 72.9642, 61.8971, 7
```

Ada juga plot statistik sederhana. Kami menggunakannya untuk menampilkan garis dan titik secara bersamaan. Alternatifnya adalah memanggil plot2d dua kali dengan >add.

```
>statplot(YT,BT1,"b"):
```



Gambar 12.371 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-019.png

Tentukan beberapa warna untuk setiap pesta.

```
>CP:=[rgb(0.5,0.5,0.5),red,yellow,green,rgb(0.8,0,0)];
```

Sekarang kita bisa memplot hasil pemilu 2009 dan perubahannya menjadi satu plot dengan menggunakan gambar. Kita dapat menambahkan vektor kolom ke setiap plot.

```
>figure(2,1); ...
```

```
> figure(1); columnsplot(BW[6,3:7],P,color=CP); ...
```

```
> figure(2); columnsplot(BW[6,3:7]-BW[5,3:7],P,color=CP); ...
```

```
> figure(0):
```

Plot data menggabungkan deretan data statistik dalam satu plot.

```
>J:=BW[,1]'; DP:=BW[,3:7]'; ...
```

```
> dataplot(YT,BT',color=CP); ...
```

```
> labelbox(P,colors=CP,styles="[]",>points,w=0.2,x=0.3,y=0.4):
```

Plot kolom 3D memperlihatkan baris data statistik dalam bentuk kolom. Kami memberikan label untuk baris dan kolom. sudut adalah sudut pandang.

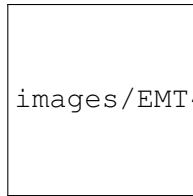


Gambar 12.372 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-020.png



Gambar 12.373 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-021.png

```
> columnsplot3d(BT,scols=P,srows=YT, ...  
> angle=30°,ccols=CP):
```



Gambar 12.374 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-022.png

Representasi lainnya adalah plot mosaik. Perhatikan bahwa kolom plot mewakili kolom matriks di sini. Karena panjang label CDU/CSU, kami mengambil jendela yang lebih kecil dari biasanya.

```
> shrinkwindow(>smaller); ...  
> mosaicplot(BT',srows=YT,scols=P,color=CP,style="#"); ...  
> shrinkwindow():
```

Kita juga bisa membuat diagram lingkaran. Karena hitam dan



images/EMT4Statistika_Shintia Nurcholisa

Gambar 12.375 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-023.png

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/EMT4Statistika_Shintia Nurcholisa

Gambar 12.376 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-024.png

Ini adalah jenis plot lainnya.

```
>starplot(normal(1,10)+4,lab=1:10,>rays):
```



images/EMT4Statistika_Shintia Nurcholisa

Gambar 12.377 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-025.png

Beberapa plot di plot2d bagus untuk statika. Berikut adalah plot impuls dari data acak, terdistribusi secara seragam di [0,1].

```
>plot2d(makeimpulse(1:10,random(1,10)),>bar):
```

Namun untuk data yang terdistribusi secara eksponensial, kita

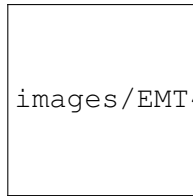


images/EMT4Statistika_Shintia Nurch

Gambar 12.378 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-026.png

mungkin memerlukan plot logaritmik.

```
>logimpulseplot(1:10,-log(random(1,10))*10):
```



images/EMT4Statistika_Shintia Nurch

Gambar 12.379 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-027.png

Fungsi Columnplot() lebih mudah digunakan, karena hanya memerlukan vektor nilai. Selain itu, ia dapat mengatur labelnya ke apa pun yang kita inginkan, kami telah mendemonstrasikannya di tutorial ini.

Ini adalah aplikasi lain, di mana kita menghitung karakter dalam sebuah kalimat dan membuat statistik.

```
>v=strtochar("the quick brown fox jumps over the lazy dog");
...
> w=ascii("a"):ascii("z"); x=getmultiplicities(w,v); ...
> cw=[]; for k=w; cw=cw|char(k); end; ...
> columnsplo(x,lab=cw,width=0.05):
```

Dimungkinkan juga untuk mengatur sumbu secara manual.

```
>n=10; p=0.4; i=0:n; x=bin(n,i)*pi*(1-p)(n-i); ...
> columnsplo(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); ...
```




images/EMT4Statistika_Shintia Nurcholis

Gambar 12.380 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-028.png

```
> textbox(["Binomial distribution", "with p=0.4"]):
```



images/EMT4Statistika_Shintia Nurcholis

Gambar 12.381 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-029.png

Berikut ini cara memplot frekuensi bilangan dalam suatu vektor.

Kami membuat vektor bilangan acak bilangan bulat 1 hingga 6.

```
>v:=inrandom(1,10,10)
```

```
[8, 5, 8, 8, 6, 8, 8, 3, 5, 5]
```

Kemudian ekstrak nomor unik di v.

```
>vu:=unique(v)
```

```
[3, 5, 6, 8]
```

Dan plot frekuensi dalam plot kolom.

```
>columnplot(getmultiplicities(vu,v),lab=vu,style="/"):
```

Kami ingin mendemonstrasikan fungsi distribusi nilai empiris.

```
>x=normal(1,20);
```

Fungsi `empdist(x,vs)` memerlukan array nilai yang diurutkan.

Jadi kita harus mengurutkan x sebelum kita dapat menggunakannya.



images/EMT4Statistika_Shintia Nurch

Gambar 12.382 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-030.png

```
>xs=sort(x);
```

Kemudian kita plot distribusi empiris dan beberapa batang kepadatan ke dalam satu plot. Alih-alih plot batang untuk distribusi kali ini kami menggunakan plot gigi gergaji.

```
>figure(2,1); ...
```

```
> figure(1); plot2d("empdist",-4,4;xs); ...
```

```
> figure(2); plot2d(histo(x,v=-4:0.2:4,<bar)); ...
```

```
> figure(0):
```



images/EMT4Statistika_Shintia Nurch

Gambar 12.383 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-031.png

Plot sebar mudah dilakukan di Euler dengan plot titik biasa. Grafik berikut menunjukkan bahwa X dan $X+Y$ jelas berkorelasi positif.

```
>x=normal(1,100); plot2d(x,x+rotright(x),>points,style=".."):
```

Seringkali kita ingin membandingkan dua sampel dengan distribusi yang berbeda. Hal ini dapat dilakukan dengan plot kuantil-kuantil.

Untuk pengujiannya, kami mencoba distribusi student-t dan distribusi eksponensial.



images/EMT4Statistika_Shintia Nurcholis

Gambar 12.384 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-032.png

```
>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/EMT4Statistika_Shintia Nurcholis

Gambar 12.385 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-033.png

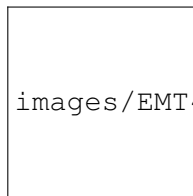
Plot tersebut dengan jelas menunjukkan bahwa nilai terdistribusi normal cenderung lebih kecil di ujung ekstrim.

Jika kita mempunyai dua distribusi yang ukurannya berbeda, kita dapat memperluas distribusi yang lebih kecil atau mengecilkan distribusi yang lebih besar. Fungsi berikut ini baik untuk keduanya. Dibutuhkan nilai median dengan persentase antara 0 dan 1.

```
>function medianexpand (x,n) := median(x,p=linspace(0,1,n-1));
```

Mari kita bandingkan dua distribusi yang sama.

```
>x=random(1000); y=random(400); ...
> plot2d("x",0,1,style="--"); ...
> plot2d(sort(medianexpand(x,400)),sort(y),>points,color=red,style="x",>add):
```



Gambar 12.386 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-034.png

REGRESI DAN KORELASI

Regresi linier dapat dilakukan dengan fungsi `polyfit()` atau berbagai fungsi fit.

Sebagai permulaan kita menemukan garis regresi untuk data univariat dengan `polyfit(x,y,1)`.

```
>x=1:10; y=[2,3,1,5,6,3,7,8,9,8]; writetable(x'|y',labc=["x","y"])
```

x	y
1	2
2	3
3	1
4	5
5	6
6	3
7	7
8	8
9	9
10	8

Kami ingin membandingkan kecocokan yang tidak berbobot dan berbobot. Pertama koefisien kecocokan linier.

```
>p=polyfit(x,y,1)
```

```
[0.733333, 0.812121]
```

Sekarang koefisien dengan bobot yang menekankan nilai terakhir.

```
>w &= "exp(-(x-10)^2/10)"; pw=polyfit(x,y,1,w=w(x))
```

```
[4.71566, 0.38319]
```

Kami memasukkan semuanya ke dalam satu plot untuk titik dan garis regresi, dan untuk bobot yang digunakan.

```

>figure(2,1); ...
> figure(1); statplot(x,y,"b",xl="Regression"); ...
> plot2d("evalpoly(x,p)",>add,color=blue,style="-"); ...
> plot2d("evalpoly(x,pw)",5,10,>add,color=red,style="-"); ...
> figure(2); plot2d(w,1,10,>filled,style="f",fillcolor=red,xl=w); ...
> figure(0):

```



images/EMT4Statistika_Shintia Nurcholisa

Gambar 12.387 images/EMT4Statistika.Shintia%20Nurcholisa.23030630071-035.png

Contoh lain kita membaca survei siswa, usia mereka, usia orang tua mereka dan jumlah saudara kandung dari sebuah file.

Tabel ini berisi “m” dan “f” di kolom kedua. Kami menggunakan variabel tok2 untuk mengatur terjemahan yang tepat alih-alih membiarkan readtable() mengumpulkan terjemahannya.

```

>{MS,hd}:=readtable("table1.dat",tok2:["m","f"]); ...
> writetable(MS,labc=hd,tok2:["m","f"]);

```

```

Could not open the file
table1.dat
for reading!

```

```

Try "trace errors" to inspect local variables after errors
readtable:

```

```

    if filename!=none then open(filename,"r"); endif;

```

Bagaimana usia bergantung satu sama lain? Kesan pertama muncul dari plot sebar berpasangan.

```

>scatterplots(tablecol(MS,3:5),hd[3:5]):

```

```

Variable or function MS not found.

```

```

Error in:

```

```

scatterplots(tablecol(MS,3:5),hd[3:5]): ...

```

Jelas terlihat bahwa usia ayah dan ibu saling bergantung satu sama lain. Mari kita tentukan dan plot garis regresinya.

```
>cs:=MS[,4:5]'; ps:=polyfit(cs[1],cs[2],1)
```

```
MS is not a variable!
```

```
Error in:
```

```
cs:=MS[,4:5]'; ps:=polyfit(cs[1],cs[2],1) ...
```

Ini jelas merupakan model yang salah. Garis regresinya adalah $s=17+0,74t$, dengan t adalah umur ibu dan s adalah umur ayah. Perbedaan usia mungkin sedikit bergantung pada usia, tapi tidak terlalu banyak.

Sebaliknya, kami mencurigai fungsi seperti $s=a+t$. Maka a adalah mean dari $s-t$. Ini adalah perbedaan usia rata-rata antara ayah dan ibu.

```
>da:=mean(cs[2]-cs[1])
```

```
cs is not a variable!
```

```
Error in:
```

```
da:=mean(cs[2]-cs[1]) ...
```

Mari kita plot ini menjadi satu plot sebar.

```
>plot2d(cs[1],cs[2],>points); ...
```

```
> plot2d("evalpoly(x,ps)",color=red,style=".",>add); ...
```

```
> plot2d("x+da",color=blue,>add):
```

```
cs is not a variable!
```

```
Error in:
```

```
plot2d(cs[1],cs[2],>points); plot2d("evalpoly(x,p
```

Berikut adalah plot kotak dari dua zaman tersebut. Ini hanya menunjukkan, bahwa usianya berbeda-beda.

```
>boxplot(cs,["mothers","fathers"]):
```



```
Variable or function cs not found.
Error in:
boxplot(cs, ["mothers", "fathers"]): ...
      ^
```

Menariknya, perbedaan median tidak sebesar perbedaan mean.

```
>median(cs[2])-median(cs[1])
```

```
cs is not a variable!
Error in:
median(cs[2])-median(cs[1]) ...
      ^
```

Koefisien korelasi menunjukkan korelasi positif.

```
>correl(cs[1],cs[2])
```

```
cs is not a variable!
Error in:
correl(cs[1],cs[2]) ...
      ^
```

Korelasi pangkat merupakan ukuran keteraturan yang sama pada kedua vektor. Hal ini juga cukup positif.

```
>rankcorrel(cs[1],cs[2])
```

```
cs is not a variable!
Error in:
rankcorrel(cs[1],cs[2]) ...
      ^
```

MEMBUAT FUNGSI BARU

Tentu saja, bahasa EMT dapat digunakan untuk memprogram fungsi-fungsi baru. Misalnya, kita mendefinisikan fungsi skewness.

$$sk(x) = \frac{\sqrt{n} \sum_i (x_i - m)^3}{(\sum_i (x_i - m)^2)^{3/2}}$$

dimana 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
```

Seperti yang Anda lihat, kita dapat dengan mudah menggunakan bahasa matriks untuk mendapatkan implementasi yang sangat singkat dan efisien. Mari kita coba fungsi ini.

```
>data=normal(20); skew(normal(10))  
  
-0.198710316203
```

Berikut adalah fungsi lainnya, yang disebut koefisien skewness Pearson.

```
>function skew1 (x) := 3*(mean(x)-median(x))/dev(x)  
>skew1(data)  
  
-0.0801873249135
```


SIMULASI MONTE CARLO

Euler dapat digunakan untuk mensimulasikan kejadian acak. Kita telah melihat contoh sederhana di atas. Ini satu lagi, yang mensimulasikan 1000 kali lemparan 3 dadu, dan menanyakan pembagian jumlahnya.

```
>ds:=sum(intrandom(1000,3,6))'; fs=getmultiplicities(3:18,ds)
```

```
[5, 17, 35, 44, 75, 97, 114, 116, 143, 116,  
22, 13, 6]
```

Kita bisa merencanakannya sekarang.

```
>columnplot(fs,lab=3:18):
```



images/EMT4Statistika_Shintia Nurch

Gambar 12.388 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-037.png

Untuk menentukan distribusi yang diharapkan tidaklah mudah. Kami menggunakan rekursi tingkat lanjut untuk ini.

Fungsi berikut menghitung banyaknya cara bilangan k dapat direpresentasikan sebagai jumlah dari 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>=1 && k<=m  
else  
    sum=0;  
    loop 1 to m; sum=sum+countways (k-#,n-1,m); end;  
    return sum;  
end;  
endfunction
```

Berikut ini hasil dari tiga kali lemparan dadu.

```
>countways(5:25,5,5)
```

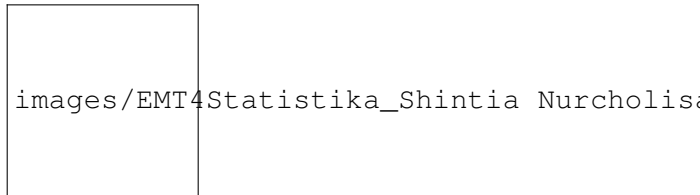
```
[1, 5, 15, 35, 70, 121, 185, 255, 320, 365, 381,
255, 185, 121, 70, 35, 15, 5, 1]
```

```
>cw=countways(3:18,3,6)
```

```
[1, 3, 6, 10, 15, 21, 25, 27, 27, 25, 21, 15,
1]
```

Kami menambahkan nilai yang diharapkan ke plot.

```
>plot2d(cw/6^3*1000,>add); plot2d(cw/6^3*1000,>points,>add):
```



Gambar 12.389 images/EMT4Statistika_Shintia%20Nurcholis_23030630071-038.png

Untuk simulasi lain, deviasi nilai rata-rata dari n variabel acak berdistribusi normal 0-1 adalah $1/\sqrt{n}$.

```
>longformat; 1/sqrt(10)
```

```
0.316227766017
```

Mari kita periksa ini dengan simulasi. Kita hasilkan 10000 kali 10 vektor acak.


```
>M=normal(10000,10); dev(mean(M)')
```

```
0.319493614817
```

```
>plot2d(mean(M)',>distribution):
```

Median dari 10 bilangan acak berdistribusi normal 0-1 memiliki deviasi yang lebih besar.

```
>dev(median(M)')
```



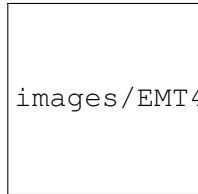
images/EMT4Statistika_Shintia Nurch

Gambar 12.390 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-039.png

0.374460271535

Karena kita dapat dengan mudah menghasilkan lintasan acak, kita dapat mensimulasikan proses Wiener. Kita mengambil 1000 langkah dari 1000 proses. Kemudian kita memetakan deviasi standar dan rata-rata langkah ke- n dari proses ini bersama dengan nilai yang diharapkan dalam warna merah.

```
>n=1000; m=1000; M=cumsum(normal(n,m)/sqrt(m)); ...  
> t=(1:n)/n; figure(2,1); ...  
> figure(1); plot2d(t,mean(M')'); plot2d(t,0,color=red,>add); ...  
> figure(2); plot2d(t,dev(M')'); plot2d(t,sqrt(t),color=red,>add); ...  
> figure(0):
```



images/EMT4Statistika_Shintia Nurch

Gambar 12.391 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-040.png

PENGUJIAN

Pengujian merupakan alat penting dalam statistik. Dalam Euler, banyak pengujian yang diterapkan. Semua pengujian ini menghasilkan galat yang kita terima jika kita menolak hipotesis nol.

Sebagai contoh, kita menguji lemparan dadu untuk distribusi seragam. Pada 600 lemparan, kita memperoleh nilai berikut, yang kita masukkan ke dalam uji chi-kuadrat.

```
>chitest([90,103,114,101,103,89],dup(100,6)')
```

```
0.498830517952
```

Uji chi-square juga memiliki modus, yang menggunakan simulasi Monte Carlo untuk menguji statistik. Hasilnya harus hampir sama. Parameter >p menginterpretasikan vektor y sebagai vektor probabilitas.

```
>chitest([90,103,114,101,103,89],dup(1/6,6)',>p,>montecarlo)
```

```
0.526
```

Kesalahan ini terlalu besar. Jadi kita tidak dapat menolak distribusi seragam. Ini tidak membuktikan bahwa dadu kita adil. Namun, kita tidak dapat menolak hipotesis kita.

Selanjutnya, kita menghasilkan 1000 lemparan dadu menggunakan generator angka acak, dan melakukan pengujian yang sama.

```
>n=1000; t=random([1,n*6]); chitest(count(t*6,6),dup(n,6)')
```

```
0.528028118442
```

Mari kita uji nilai rata-rata 100 dengan uji-t.

```
>s=200+normal([1,100])*10; ...
```

```
> ttest(mean(s),dev(s),100,200)
```

```
0.0218365848476
```


Fungsi `ttest()` memerlukan nilai rata-rata, deviasi, jumlah data, dan nilai rata-rata yang akan diuji.

Sekarang mari kita periksa dua pengukuran untuk nilai rata-rata yang sama. Kita tolak hipotesis bahwa keduanya memiliki nilai rata-rata yang sama, jika hasilnya $< 0,05$.

```
>tcomparedata(normal(1,10),normal(1,10))
```

```
0.38722000942
```

Jika kita menambahkan bias pada satu distribusi, kita akan mendapatkan lebih banyak penolakan. Ulangi simulasi ini beberapa kali untuk melihat efeknya.

```
>tcomparedata(normal(1,10),normal(1,10)+2)
```

```
5.60009101758e-07
```

Pada contoh berikutnya, kita buat 20 lemparan dadu acak sebanyak 100 kali dan hitung angka-angka yang ada di dalamnya. Rata-rata harus ada $20/6=3,3$ angka.

```
>R=random(100,20); R=sum(R*6<=1)'; mean(R)
```

```
3.28
```

Sekarang kita bandingkan jumlah angka satu dengan distribusi binomial. Pertama kita gambarkan distribusi angka satu.

```
>plot2d(R,distribution=max(R)+1,even=1,style="\/"): 
```



images/EMT4Statistika_Shintia Nurcholisa

Gambar 12.392 images/EMT4Statistika.Shintia%20Nurcholisa.23030630071-041.png

```
>t=count(R,21);
```

Lalu kami hitung nilai yang diharapkan.

```
>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-kuadrat menolak hipotesis bahwa distribusi kami adalah distribusi binomial, jika hasilnya $< 0,05$.

```
>chitest(t1,b1)
```

```
0.53921579764
```

Contoh berikut berisi hasil dari dua kelompok orang (misalnya pria dan wanita) yang memilih satu dari enam partai.

```
>A=[23,37,43,52,64,74;27,39,41,49,63,76]; ...
```

```
> writetable(A,wc=6,labr=["m","f"],labc=1:6)
```

	1	2	3	4	5	6
m	23	37	43	52	64	74
f	27	39	41	49	63	76

Kami ingin menguji independensi suara dari jenis kelamin. Uji tabel χ^2 melakukan hal ini. Hasilnya terlalu besar untuk menolak independensi. Jadi, kami tidak dapat mengatakan, apakah pemungutan suara bergantung pada jenis kelamin dari data ini.

```
>tabletest(A)
```

```
0.990701632326
```

Berikut ini adalah tabel yang diharapkan, jika kita mengasumsikan frekuensi pemungutan suara yang diamati.

```
>writetable(expectedtable(A),wc=6,dc=1,labr=["m","f"],labc=1:6)
```

	1	2	3	4	5	6
m	24.9	37.9	41.9	50.3	63.3	74.7
f	25.1	38.1	42.1	50.7	63.7	75.3

Kita dapat menghitung koefisien kontingensi yang dikoreksi. Karena sangat mendekati 0, kita simpulkan bahwa pemungutan suara tidak bergantung pada jenis kelamin.

>contingency(A)

0.0427225484717

BEBERAPA PENGUJIAN LAINNYA

Selanjutnya, kami menggunakan analisis varians (uji F) untuk menguji tiga sampel data berdistribusi normal untuk nilai rata-rata yang sama. Metode ini disebut ANOVA (analisis varians). Dalam Euler, fungsi `varanalysis()` digunakan.

```
>x1=[109,111,98,119,91,118,109,99,115,109,94]; mean(x1),  
106.545454545
```

```
>x2=[120,124,115,139,114,110,113,120,117]; mean(x2),  
119.111111111
```

```
>x3=[120,112,115,110,105,134,105,130,121,111]; mean(x3)  
116.3
```

```
>varanalysis(x1,x2,x3)  
0.0138048221371
```

Artinya, kita menolak hipotesis nilai rata-rata yang sama. Kita melakukan ini dengan probabilitas kesalahan sebesar 1,3%.

Ada juga uji median, yang menolak sampel data dengan distribusi rata-rata yang berbeda dengan menguji median sampel gabungan.

```
>a=[56,66,68,49,61,53,45,58,54];  
>b=[72,81,51,73,69,78,59,67,65,71,68,71];  
>mediantest(a,b)  
0.0241724220052
```

Uji kesetaraan lainnya adalah uji peringkat. Uji peringkat jauh lebih tajam daripada uji median.

```
>ranktest(a,b)
```

0.00199969612469

Dalam contoh berikut, kedua distribusi memiliki rata-rata yang sama.

```
>ranktest(random(1,100),random(1,50)*3-1)
```

0.129608141484

Sekarang, mari kita coba simulasikan dua perawatan a dan b yang diterapkan pada orang yang berbeda.

```
>a=[8.0,7.4,5.9,9.4,8.6,8.2,7.6,8.1,6.2,8.9];
```

```
>b=[6.8,7.1,6.8,8.3,7.9,7.2,7.4,6.8,6.8,8.1];
```

Uji signum memutuskan, apakah a lebih baik dari b.

```
>signtest(a,b)
```

0.0546875

Ini adalah kesalahan yang sangat besar. Kita tidak dapat menolak bahwa a sama baiknya dengan b.

Uji Wilcoxon lebih tajam daripada uji ini, tetapi bergantung pada nilai kuantitatif perbedaannya.

```
>wilcoxon(a,b)
```

0.0296680599405

Mari kita coba dua pengujian lagi menggunakan seri yang dihasilkan.

```
>wilcoxon(normal(1,20),normal(1,20)-1)
```

0.0068706451766

```
>wilcoxon(normal(1,20),normal(1,20))
```

0.275145971064

ANGKA ACAK

Berikut ini adalah pengujian untuk generator angka acak. Euler menggunakan generator yang sangat bagus, jadi kita tidak perlu mengharapkan masalah apa pun.

Pertama-tama kita menghasilkan sepuluh juta angka acak dalam $[0,1]$.

```
>n:=10000000; r:=random(1,n);
```

Berikutnya kita hitung jarak antara dua angka kurang dari 0,05.

```
>a:=0.05; d:=differences(nonzeros(r<a));
```

Terakhir, kami memplot berapa kali setiap jarak terjadi, dan membandingkannya dengan nilai yang diharapkan.

```
>m=getmultiplicities(1:100,d); plot2d(m); ...  
> plot2d("n*(1-a)(x-1)*a2",color=red,>add):
```



images/EMT4Statistika_Shintia Nurch

Gambar 12.393 images/EMT4Statistika_Shintia%20Nurcholisa.23030630071-042.png

Hapus data.

```
>remvalue n;
```


PENDAHULUAN BAGI PENGGUNA PROYEK R

Jelas, EMT tidak bersaing dengan R sebagai paket statistik. Akan tetapi, ada banyak prosedur dan fungsi statistik yang tersedia di EMT juga. Jadi, EMT dapat memenuhi kebutuhan dasar. Lagi pula, EMT dilengkapi dengan paket numerik dan sistem aljabar komputer.

Buku catatan ini ditujukan bagi Anda yang sudah familier dengan R, tetapi perlu mengetahui perbedaan sintaksis EMT dan R. Kami mencoba memberikan gambaran umum tentang hal-hal yang jelas dan kurang jelas yang perlu Anda ketahui.

Selain itu, kami melihat cara untuk bertukar data antara kedua sistem.

Harap dicatat bahwa ini adalah pekerjaan yang masih dalam tahap pengerjaan.

SINTAKSIS DASAR

Hal pertama yang Anda pelajari di R adalah membuat vektor. Dalam EMT, perbedaan utamanya adalah operator : dapat mengambil ukuran langkah. Selain itu, operator ini memiliki daya pengikatan 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, 7, 7.5, 8, 8.5, 9]
```

Fungsi c() tidak ada. Dimungkinkan untuk menggunakan vektor guna menggabungkan berbagai hal.

Contoh berikut ini, seperti banyak contoh lainnya, berasal dari “Interoduction to R” yang disertakan dalam proyek R. Jika Anda membaca PDF ini, Anda akan menemukan bahwa saya mengikuti alurnya dalam tutorial ini.

```
>x=[10.4, 5.6, 3.1, 6.4, 21.7]; [x,0,x]
```

```
[10.4, 5.6, 3.1, 6.4, 21.7, 0, 10.4, 5.6, 3.1,
```

Operator titik dua dengan ukuran langkah EMT digantikan oleh fungsi seq() di R. Kita dapat menulis fungsi ini dalam EMT.

```
>function seq(a,b,c) := a:b:c; ...
```

```
> seq(0,-0.1,-1)
```

```
[0, -0.1, -0.2, -0.3, -0.4, -0.5, -0.6, -0.7,
```

Fungsi rep() dari R tidak ada dalam 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
```

Perhatikan bahwa “=” atau “:=” digunakan untuk penugasan. Operator “->” digunakan untuk unit dalam EMT.

```
>125km -> " miles"
```

```
77.6713990297 miles
```

Operator “<-” untuk penugasan menyematkan dan bukan ide yang baik untuk R. Berikut ini akan membandingkan a dan -4 dalam EMT.

```
>a=2; a<-4
```

```
0
```

Dalam R, “a<-4<3” berfungsi, tetapi “a<-4<-3” tidak. Saya juga mengalami ambiguitas serupa dalam EMT, tetapi mencoba menghilangkannya sedikit demi sedikit.

EMT dan R memiliki vektor bertipe boolean. Namun dalam EMT, angka 0 dan 1 digunakan untuk mewakili false dan true. Dalam R, nilai true dan false tetap dapat digunakan dalam aritmatika biasa seperti dalam EMT.

```
>x<5, %*x
```

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

```
[0, 0, 3.1, 0, 0]
```

EMT memunculkan kesalahan atau menghasilkan NAN, tergantung pada tanda “kesalahan”.

```
>errors off; 0/0, isNAN(sqrt(-1)), errors on;
```

```
NAN
```

```
1
```

String sama di R dan EMT. Keduanya berada di lokal saat ini, bukan di Unicode.

Di R ada paket untuk Unicode. Di EMT, string dapat berupa string Unicode. String unicode dapat diterjemahkan ke pengodean lokal dan sebaliknya. Selain itu, u”...” dapat berisi entitas HTML.

```
>u”© Ren&eacute; Grothmann”
```

Berikut ini mungkin atau mungkin tidak ditampilkan dengan benar pada sistem Anda sebagai A dengan titik dan garis di atasnya. Hal ini bergantung pada font yang Anda gunakan.

```
>chartoutf([480])
```

Penggabungan string dilakukan dengan “+” atau “|”. String dapat menyertakan angka, yang akan dicetak dalam format saat ini.

```
>“pi=”+pi
```

```
pi = 3.14159265359
```


PENGINDEKSAN

Sering kali, ini akan berfungsi seperti di R.

Namun EMT akan menginterpretasikan indeks negatif dari belakang vektor, sementara R menginterpretasikan $x[n]$ sebagai x tanpa elemen ke- n .

```
>x, x[1:3], x[-2]

[10.4,  5.6,  3.1,  6.4,  21.7]
[10.4,  5.6,  3.1]
6.4
```

Perilaku R dapat dicapai dalam EMT dengan `drop()`.

```
>drop(x,2)

[10.4,  3.1,  6.4,  21.7]
```

Vektor logika tidak diperlakukan secara berbeda sebagai indeks dalam EMT, berbeda dengan R. Anda perlu mengekstrak elemen bukan nol terlebih dahulu dalam EMT.

```
>x, x>5, x[nonzeros(x>5)]

[10.4,  5.6,  3.1,  6.4,  21.7]
[1,  1,  0,  1,  1]
[10.4,  5.6,  6.4,  21.7]
```

Sama seperti di R, vektor indeks dapat berisi pengulangan.

```
[10.4,  5.6,  5.6,  10.4]
```

Namun, nama untuk indeks tidak dimungkinkan dalam EMT. Untuk paket statistik, hal 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)]; ... ...
      ^
```

```
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 terdapat vektor yang terus bertambah. Anda dapat menetapkan vektor numerik kosong `v` dan menetapkan nilai ke elemen `v[17]`. Hal ini tidak mungkin dilakukan di EMT.

Berikut ini agak tidak efisien.

```
>v=[]; for i=1 to 10000; v=v[i]; end;
```

EMT sekarang akan membuat vektor dengan `v` dan `i` yang ditambahkan pada tumpukan dan menyalin vektor itu kembali ke variabel global `v`.

Yang lebih efisien mendefinisikan vektor terlebih dahulu.

```
>v=zeros(10000); for i=1 to 10000; v[i]=i; end;
```

Yang lebih efisien mendefinisikan vektor terlebih dahulu.

```
>complex(1:4)
```

```
[ 1+0i , 2+0i , 3+0i , 4+0i ]
```

Konversi ke string hanya dimungkinkan untuk tipe data dasar. Format saat ini digunakan untuk penggabungan string sederhana. Namun, 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
```

```
>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 keluaran.

```
>convertmxm(1:10)
```

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

Untuk Latex perintah `tex` dapat digunakan untuk mendapatkan perintah Latex.

```
>tex(&[1,2,3])
```

```
\left[ 1 , 2 , 3 \right]
```

FAKTOR DAN TABEL

Dalam pengantar R terdapat contoh dengan apa yang disebut faktor.

Berikut ini adalah daftar wilayah dari 30 negara bagian.

```
>austates = [“tas”, “sa”, “qld”, “nsw”, “nsw”, “nt”, “wa”, “wa”,  
...  
> “qld”, “vic”, “nsw”, “vic”, “qld”, “qld”, “sa”, “tas”, ...  
> “sa”, “nt”, “wa”, “vic”, “qld”, “nsw”, “nsw”, “wa”, ...  
> “sa”, “act”, “nsw”, “vic”, “vic”, “act”];
```

Asumsikan, kita memiliki pendapatan yang sesuai di setiap negara bagian.

```
>incomes = [60, 49, 40, 61, 64, 60, 59, 54, 62, 69, 70, 42, 56,  
...  
> 61, 61, 61, 58, 51, 48, 65, 49, 49, 41, 48, 52, 46, ...  
> 59, 46, 58, 43];
```

Sekarang, kita ingin menghitung rata-rata pendapatan di wilayah tersebut. Sebagai program statistik, R memiliki `factor()` dan `tapply()` untuk ini.

EMT dapat melakukan ini 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,  
8, 7, 4, 2, 2, 8, 5, 1, 2, 7, 7, 1]
```

Pada titik tersebut, kita dapat menulis fungsi loop kita sendiri untuk melakukan sesuatu hanya untuk satu faktor.

Atau kita dapat meniru fungsi `tapply()` dengan cara berikut.

```
>function map_tappl (i; f$:call, cat, x) ...  
  
u=sort(unique(cat));  
f=indexof(u,cat);  
return f$(x[nonzeros(f==indexof(u,i))]);  
endfunction
```

Agak tidak efisien, karena menghitung wilayah unik untuk setiap i, tetapi berhasil.

```
>tappl(auterr,"mean",austates,incomes)
```

```
[44.5, 57.3333333333, 55.5, 53.6, 55, 60.5, 56, 52.25]
```

Perhatikan bahwa ini berfungsi untuk setiap vektor wilayah.

```
>tappl(["act","nsw"],"mean",austates,incomes)
```

```
[44.5, 57.3333333333]
```

Sekarang, paket statistik EMT mendefinisikan tabel seperti di R. Fungsi readtable() dan writetable() dapat digunakan untuk input dan output.

Jadi kita dapat mencetak pendapatan negara rata-rata di wilayah dengan cara yang mudah.

```
>writetable(tappl(auterr,"mean",austates,incomes),labc=auterr,wc=7)
```

act	nsw	nt	qld	sa	tas	vic	wa
44.5	57.33	55.5	53.6	55	60.5	56	52.25

Kita juga dapat mencoba meniru perilaku R sepenuhnya.

Faktor-faktor tersebut harus disimpan dalam suatu koleksi dengan jenis dan kategori (negara bagian dan teritori dalam contoh kita). Untuk EMT, kita tambahkan indeks yang telah dihitung sebelumnya.

```
>function makef(t)...
```

```
## Factor data
```

```
## Returns a collection with data t, unique data, indices.
```

```
## See: tapply
```

```
u=sort(unique(t));
```

```
return ({t,u,indexofsorted(u,t)});
```

```
endfunction
```

```
>statef=makef(austates);
```

Sekarang elemen ketiga dari koleksi akan berisi indeks.

```
>statef[3]
```

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

Sekarang kita dapat meniru `tapply()` dengan cara berikut. Fungsi ini akan mengembalikan tabel sebagai kumpulan data tabel dan judul kolom.

```
>function tapply (t:vector,tf,f$:call) ...
```

```
## Makes a table of data and factors
## tf : output of makef()
## See: makef
uf=tf[2]; f=tf[3]; x=zeros(length(uf));
for i=1 to length(uf);
    ind=nonzeros(f==i);
    if length(ind)==0 then x[i]=NAN;
    else x[i]=f$(t[ind]);
endif;
end;
return {{x,uf}};
endfunction
```

Kami tidak menambahkan banyak pemeriksaan tipe di sini. Satu-satunya tindakan pencegahan menyangkut kategori (faktor) tanpa data. Namun, seseorang harus memeriksa panjang `t` yang benar dan kebenaran koleksi `tf`.

Tabel ini dapat dicetak sebagai tabel dengan `writetable()`.

```
>writetable(tapply(incomes,statef,"mean"),wc=7)
```

act	nsw	nt	qld	sa	tas	vic	
44.5	57.33	55.5	53.6	55	60.5	56	52

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 di R.

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

2	3
7	8
12	13
17	18

Namun, dalam R dimungkinkan untuk menetapkan daftar indeks vektor tertentu ke suatu nilai. Hal yang sama dimungkinkan dalam 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
```

Kami mendemonstrasikan ini untuk menunjukkan bahwa matriks dilewatkan dengan referensi dalam EMT. Jika Anda tidak

ingin mengubah matriks asli M, Anda perlu menyalinnya dalam fungsi tersebut.

```
>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

Produk luar dalam EMT hanya dapat dilakukan antara vektor. Hal ini dilakukan secara otomatis karena bahasa matriks. Satu vektor harus berupa vektor kolom dan yang lainnya berupa vektor baris.

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

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

Dalam pengantar PDF untuk R terdapat sebuah contoh, yang menghitung distribusi ab-cd untuk a,b,c,d yang dipilih secara acak dari 0 hingga n. Solusi dalam R adalah membentuk matriks 4 dimensi dan menjalankan table() di atasnya.

Tentu saja, ini dapat dicapai dengan loop. Namun, loop tidak efektif dalam EMT atau R. Dalam EMT, kita dapat menulis loop dalam C dan itu akan menjadi solusi tercepat.

Namun, kita ingin meniru perilaku R. Untuk ini, kita perlu meratakan perkalian ab dan membuat matriks ab-cd.

```
>a=0:6; b=a'; p=flatten(a*b); q=flatten(p-p'); ...
> u=sort(unique(q)); f=getmultiplicities(u,q); ...
> statplot(u,f,"h"):
```

Selain multiplisitas yang tepat, EMT dapat menghitung frekuensi dalam vektor.

```
>getfrequencies(q,-50:10:50)
```

```
[0, 23, 132, 316, 602, 801, 333, 141, 53, 0]
```



Gambar 12.394 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-043.png

Cara termudah untuk memplot ini sebagai distribusi adalah sebagai berikut.

```
>plot2d(q,distribution=11):
```



Gambar 12.395 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-044.png

Namun, Anda juga dapat menghitung terlebih dahulu jumlah dalam interval yang dipilih. Tentu saja, berikut ini menggunakan `getfrequencies()` secara internal.

Karena fungsi `histo()` mengembalikan frekuensi, kita perlu menskalakannya sehingga integral di bawah grafik batang adalah 1.

```
>{x,y}=histo(q,v=-55:10:55); y=y/sum(y)/differences(x); ...  
> plot2d(x,y,>bar,style="/"):
```




images/EMT4Statistika_Shintia Nurcholisa

Gambar 12.396 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-045.png

DAFTAR

EMT memiliki dua jenis daftar. Satu adalah daftar global yang dapat diubah, dan yang lainnya adalah jenis daftar yang tidak dapat diubah. Kami tidak peduli dengan daftar global di sini.

Jenis daftar yang tidak dapat diubah disebut koleksi dalam EMT. Ia berperilaku seperti struktur dalam C, tetapi elemennya hanya diberi nomor dan tidak diberi nama.

$$>L=\{\text{"Fred"}, \text{"Flintstone"}, 40, [1990, 1992]\}$$

Fred

Flintstone

40

[1990, 1992]

Saat ini unsur-unsur tersebut tidak memiliki nama, meskipun nama dapat ditetapkan untuk tujuan khusus. Unsur-unsur tersebut diakses dengan angka.

$$>(L[4])[2]$$

1992

INPUT DAN OUTPUT FILE (MEMBACA DAN MENULIS DATA)

Anda sering kali ingin mengimpor matriks data dari sumber lain ke EMT. Tutorial ini memberi tahu Anda tentang berbagai cara untuk mencapainya. Fungsi sederhana adalah `writematrix()` dan `readmatrix()`.

Mari kita tunjukkan cara membaca dan menulis vektor bilangan real ke dalam file.

```
>a=random(1,100); mean(a), dev(a),
```

```
0.49815
```

```
0.28037
```

Untuk menulis data ke dalam berkas, kami menggunakan fungsi `writematrix()`.

Karena pengantar ini kemungkinan besar berada di dalam direktori, tempat pengguna tidak memiliki akses tulis, kami menulis data ke direktori beranda pengguna. Untuk buku catatan sendiri, ini tidak diperlukan, karena berkas data akan ditulis ke dalam direktori yang sama.

```
>filename="test.dat";
```

Sekarang kita tulis vektor kolom `a'` ke dalam berkas. Ini menghasilkan satu angka di setiap baris berkas.

```
>writematrix(a',filename);
```

Untuk membaca data, kita menggunakan `readmatrix()`.

```
>a=readmatrix(filename)';
```

Dan hapus berkasnya.

```
>fileremove(filename);
```

```
>mean(a), dev(a),
```

```
0.49815
```

```
0.28037
```

Fungsi `writematrix()` atau `writetable()` dapat dikonfigurasi untuk bahasa lain.

Misalnya, jika Anda memiliki sistem bahasa Indonesia (titik desimal dengan koma), Excel Anda memerlukan nilai dengan koma desimal yang dipisahkan oleh titik koma dalam file csv (nilai default dipisahkan dengan koma). File berikut “test.csv” akan muncul di folder Anda saat ini.

```
> filename="test.csv"; ...
```

```
> writematrix(random(5,3),file=filename,separator=",");
```

Anda sekarang dapat membuka berkas ini langsung dengan Excel Indonesia.

```
> fileremove(filename);
```

Terkadang kita memiliki string dengan token seperti berikut.

```
> s1:="f m m f m m m f f f m f"; ...
```

```
> s2:="f f f m m f f";
```

Untuk menokenisasi ini, kami mendefinisikan vektor token.

```
> tok=["f","m"]
```

```
f  
m
```

Lalu kita dapat menghitung berapa kali setiap token muncul dalam string, dan memasukkan hasilnya ke dalam tabel.

```
> M:=getmultiplicities(tok,strtokens(s1))- ...
```

```
> getmultiplicities(tok,strtokens(s2));
```

Tulis tabel dengan tajuk token.

```
> writetable(M,labc=tok,labr=1:2,wc=8)
```

	f	m
1	6	7
2	5	2

Untuk statika, EMT dapat membaca dan menulis tabel.

```
> file="test.dat"; open(file,"w"); ...
```

```
> writeln("A,B,C"); writematrix(random(3,3)); ...
```

```
> close();
```

Berkasnya tampak seperti ini.

```
>printfile(file)
```

A, B, C

```
0.7003664386138074, 0.1875530821001213, 0.3262339279660.  
0.5926249243193858, 0.1522927283984059, 0.3681405830625.  
0.8065535209872989, 0.7265910840408142, 0.7332619844597.
```

Fungsi `readtable()` dalam bentuk yang paling sederhana dapat membacanya dan mengembalikan kumpulan nilai dan baris judul.

```
>L=readtable(file,>list);
```

Koleksi ini dapat dicetak dengan `writetable()` ke buku catatan, atau ke berkas.

```
>writetable(L,wc=10,dc=5)
```

A	B	C
0.70037	0.18755	0.32623
0.59262	0.15229	0.36814
0.80655	0.72659	0.73326

Matriks nilai adalah elemen pertama L. Perhatikan bahwa `mean()` dalam EMT menghitung nilai rata-rata baris matriks.

```
>mean(L[1])
```

```
0.40472  
0.37102  
0.75547
```


BERKAS CSV

Pertama, mari kita tulis matriks ke dalam berkas. Untuk output, kita buat berkas di direktori kerja saat ini.

```
>file="test.csv"; ...  
> M=random(3,3); writematrix(M,file);
```

Berikut ini isi berkas tersebut.

```
>printfile(file)
```

```
0.8221197733097619,0.821531098722547,0.77712406080940  
0.8482947121863489,0.3237767724883862,0.6501422353377  
0.1482301827518109,0.3297459716109594,0.6261901074210
```

CVS ini dapat dibuka pada sistem bahasa Inggris ke Excel dengan mengklik dua kali. Jika Anda mendapatkan berkas tersebut pada sistem bahasa Jerman, Anda perlu mengimpor data ke Excel dengan memperhatikan titik desimal.

Namun, titik desimal juga merupakan format default untuk EMT. Anda dapat membaca matriks dari berkas dengan `readmatrix()`.

```
>readmatrix(file)
```

```
0.82212    0.82153    0.77712  
0.84829    0.32378    0.65014  
0.14823    0.32975    0.62619
```

Dimungkinkan untuk menulis beberapa matriks ke dalam satu berkas. Perintah `open()` dapat membuka berkas untuk ditulis dengan parameter “w”. Nilai default untuk membaca adalah “r”.

```
>open(file,“w”); writematrix(M); writematrix(M’); close();
```

Matriks dipisahkan oleh baris kosong. Untuk membaca matriks, buka berkas dan panggil `readmatrix()` beberapa kali.

```
>open(file); A=readmatrix(); B=readmatrix(); A==B, close();
```


0	1	0
0	0	1

Di Excel atau lembar kerja serupa, Anda dapat mengekspor 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 ini contohnya.

```
>printfile("excel-data.csv")
```

```
Could not open the file
excel-data.csv
for reading!
Try "trace errors" to inspect local variables after errors
printfile:
  open(filename,"r");
```

Seperti yang Anda lihat, sistem Jerman saya menggunakan titik koma sebagai pemisah dan koma desimal. Anda dapat mengubahnya di pengaturan sistem atau di Excel, tetapi tidak diperlukan untuk membaca matriks ke EMT.

Cara termudah untuk membaca ini ke Euler adalah `readmatrix()`. Semua koma diganti dengan titik dengan parameter `>comma`. Untuk CSV bahasa Inggris, cukup abaikan parameter ini.

```
>M=readmatrix("excel-data.csv",>comma)
```

```
Could not open the file
excel-data.csv
for reading!
Try "trace errors" to inspect local variables after errors
readmatrix:
  if filename<&gt;" " then open(filename,"r"); endif;
```

Mari kita plot ini.

```
>plot2d(M'[1],M'[2:3],>points,color=[red,green]')
```

Ada cara yang lebih mendasar untuk membaca data dari sebuah berkas. Anda dapat membuka berkas dan membaca angka baris demi

Gambar 12.397 images/EMT4Statistika_Shintia%20Nurcholisa_23030630071-046.png

baris. Fungsi `getvectorline()` akan membaca angka dari sebaris data. Secara default, fungsi ini mengharapkan titik desimal. Namun, fungsi ini juga dapat menggunakan koma desimal, jika Anda memanggil `setdecimaldot(",")` sebelum menggunakan fungsi ini.

Fungsi berikut adalah contohnya. Fungsi ini akan berhenti di akhir berkas atau baris kosong.

```
>function myload (file) ...
```

```
open (file);
M=[];
repeat
    until eof();
    v=getvectorline(3);
    if length(v)>0 then M=M_v; else break; endif;
end;
return M;
close(file);
endfunction
```

```
>myload(file)
```

0.82212	0	0.82153	0	0.77712
0.84829	0	0.32378	0	0.65014
0.14823	0	0.32975	0	0.62619

Semua angka dalam berkas itu juga dapat dibaca dengan `getvector()`.

```
>open(file); v=getvector(10000); close(); redim(v[1:9],3,3)
```

0.82212	0	0.82153
0	0.77712	0.84829
0	0.32378	0

Jadi sangat mudah untuk menyimpan vektor nilai, satu nilai di setiap baris dan membaca kembali vektor ini.

```
>v=random(1000); mean(v)
```

```
0.50303
```

```
>writematrix(v',file); mean(readmatrix(file)')
```

```
0.50303
```

MENGGUNAKAN TABEL

Tabel dapat digunakan untuk membaca atau menulis data numerik. 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","three"]); ...  
> close(); ...  
> printfile(file)
```

```
one, two, three  
    0.09,      0.39,      0.86  
    0.39,      0.86,      0.71  
    0.2,       0.02,      0.83
```

Ini dapat diimpor ke Excel.

Untuk membaca berkas di EMT, kami menggunakan `readtable()`.

```
>{M,headings}=readtable(file,>clabs); ...  
> writetable(M,labc=headings)
```

```
    one      two      three  
0.09    0.39    0.86  
0.39    0.86    0.71  
0.2     0.02    0.83
```


MENGANALISIS GARIS

Anda bahkan dapat mengevaluasi setiap 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 membuat token pada baris tersebut.

```
>vt=strtokens(line)
```

```
2020-11-03
```

```
Tue
```

```
1'114.05
```

Kemudian kita dapat mengevaluasi setiap elemen garis menggunakan evaluasi yang tepat.

```
>day(vt[1]),...
```

```
>indexof(["mon","tue","wed","thu","fri","sat","sun"],tolower(vt[2])),
```

```
...
```

```
>strrepl(vt[3],"", "(")
```

```
7.3816e+05
```

```
2
```

```
1114
```

Dengan menggunakan ekspresi reguler, dimungkinkan untuk mengekstrak hampir semua informasi dari sebaris data.

Asumsikan kita memiliki baris berikut sebagai dokumen HTML.

```
>line="<tr><td>1145.45</td><td>5.6</td><td>-4.5</td><tr>"
```

```
&lt;tr&gt;&lt;td&gt;1145.45&lt;/td&gt;&lt;td&gt;5.6&lt;/td&gt;&lt;td&gt;-4.5&lt;/td&gt;&lt;/tr&gt;"
```

Untuk mengekstraknya, kami menggunakan ekspresi reguler, yang mencari

- tanda kurung tutup >,
- string apa pun yang tidak mengandung tanda kurung dengan sub-kecocokan “(...)”, tanda kurung buka dan tutup menggunakan solusi terpendek, lagi-lagi string apa pun yang tidak mengandung tanda kurung, dan tanda kurung buka <.

Ekspresi reguler agak sulit dipelajari tetapi sangat ampuh.

```
>{pos,s,vt}=strxfind(line,">([^\>]+)<.+?>([^\>]+)<");
```

Hasilnya adalah posisi kecocokan, string yang cocok, dan vektor string untuk sub-kecocokan.

```
>for k=1:length(vt); vtk, end;
```

```
1145.5
```

```
5.6
```

Berikut adalah fungsi yang membaca semua item numerik antara <td> dan </td>.

```
>function readtd (line) ...
```

```
v=[]; cp=0;
```

```
repeat
```

```
    {pos,s,vt}=strxfind(line,"<td.*?>(.+?)</td>",cp);
```

```
    until pos==0;
```

```
    if length(vt)>0 then v=v|vt[1]; endif;
```

```
    cp=pos+strlen(s);
```

```
end;
```

```
return v;
```

```
endfunction
```

```
>readtd(line+"<td>non-numerical</td>")
```

```
1145.45
```

```
5.6
```

```
-4.5
```

```
non-numerical
```

MEMBACA DARI WEB

Situs web atau berkas dengan URL dapat dibuka di EMT dan dapat dibaca baris demi baris.

Dalam contoh ini, kami membaca versi terkini dari situs EMT. Kami menggunakan ekspresi reguler untuk memindai “Versi ...” dalam judul.

```
>function readversion () ...
```

```
urlopen("http://www.euler-math-toolbox.de/Programs/Ch  
repeat  
  until urleof();  
  s=urlgetline();  
  k=strfind(s,"Version ",1);  
  if k>0 then substring(s,k,strfind(s,"<",k)-1), break  
end;  
urlclose();  
endfunction
```

```
>readversion
```

```
Version 2024-01-12
```


INPUT DAN OUTPUT VARIABEL

Anda dapat menulis variabel dalam bentuk definisi Euler ke dalam file atau ke baris perintah.

```
>writevar(pi,“mypi”);
```

```
mypi = 3.141592653589793;
```

Untuk pengujian, kami membuat file Euler di direktori kerja EMT.

```
>file=“test.e”; ...  
> writevar(random(2,2),“M”,file); ...  
> printfile(file,3)
```

```
M = [ ..  
0.5991820585590205, 0.7960280262224293;  
0.5167243983231363, 0.2996684599070898];
```

Sekarang kita dapat memuat berkas tersebut. Berkas tersebut akan mendefinisikan matriks M.

```
>load(file); show M,
```

```
M =  
0.59918    0.79603  
0.51672    0.29967
```

Ngomong-ngomong, jika `writevar()` digunakan pada suatu variabel, ia akan mencetak definisi variabel dengan nama variabel ini.

```
>writevar(M); writevar(inch$)
```

```
M = [ ..  
0.5991820585590205, 0.7960280262224293;  
0.5167243983231363, 0.2996684599070898];  
inch$ = 0.0254;
```

Kita juga dapat membuka berkas baru atau menambahkannya ke berkas yang sudah ada. Dalam contoh ini, kita menambahkannya ke berkas yang dibuat sebelumnya.

```
>open(file,"a"); ...
> writevar(random(2,2),"M1"); ...
> writevar(random(3,1),"M2"); ...
> close();
>load(file); show M1; show M2;
```

```
M1 =
  0.30287    0.15372
  0.7504    0.75401
M2 =
  0.27213
  0.053211
  0.70249
```

Untuk menghapus file apa pun gunakan `fileremove()`.

```
>fileremove(file);
```

Vektor baris dalam sebuah berkas tidak memerlukan koma, jika setiap angka berada di baris baru. Mari kita buat berkas seperti itu, tulis setiap baris satu per satu dengan `writeln()`.

```
>open(file,"w"); writeln("M = ["); ...
> for i=1 to 5; writeln(" "+random()); end; ...
> writeln("];"); close(); ...
> printfile(file)
```

```
M = [
  0.344851384551
  0.0807510017715
  0.876519562911
  0.754157709472
  0.688392638934
];
```

```
>load(file); M
```

```
[0.34485,    0.080751,    0.87652,    0.75416,    0.68839]
```

catatan : ketika mengenter perintah-perintah diatas ternyata hasil yang didapatkan berbeda-beda

LATIHAN SOAL

Nomor 1 Carilah rata-rata dan standar deviasi beserta plot dari data berikut

```
X = 2000,2500,2700,3500,4500,5000
```

```
>X=[2000,2500,2700,3500,4500,5000]; ...
```

```
> mean(X), dev(X),
```

```
3366.7
```

```
1186
```

```
>aspect(1.5); boxplot(X):
```

Nomor 2

Misalkan diberikan data skor hasil statistika dari 20 orang mahasiswa sebagai berikut:

```
70,65,79,90,60,79,86,95,100,70,60,91,68,84,59,90,88,84,86,90
```

Tentukan rata-rata dari data tersebut!

```
>X=[70,65,79,90,60,79,86,95,100,70,60,91,68,84,59,90,88,84,86,90]
```

```
[70, 65, 79, 90, 60, 79, 86, 95, 100, 70, 60, 91, 68, 84, 59, 90, 88, 84, 86, 90]
```

```
>mean(X)
```

```
79.7
```

