

Membuat Plot 2D (Kelompok 2)

Anggota Kelompok:

1. Aji Handono Warih (21301241029)
2. Afrilia Sefani (24030130024)
3. Rahma Amalia (24030130028)
4. Princess Velaninta Audrey Sava (24030130075)

Rahma Amalia (24030130028)

Fungsi dalam Satu Parameter

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

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

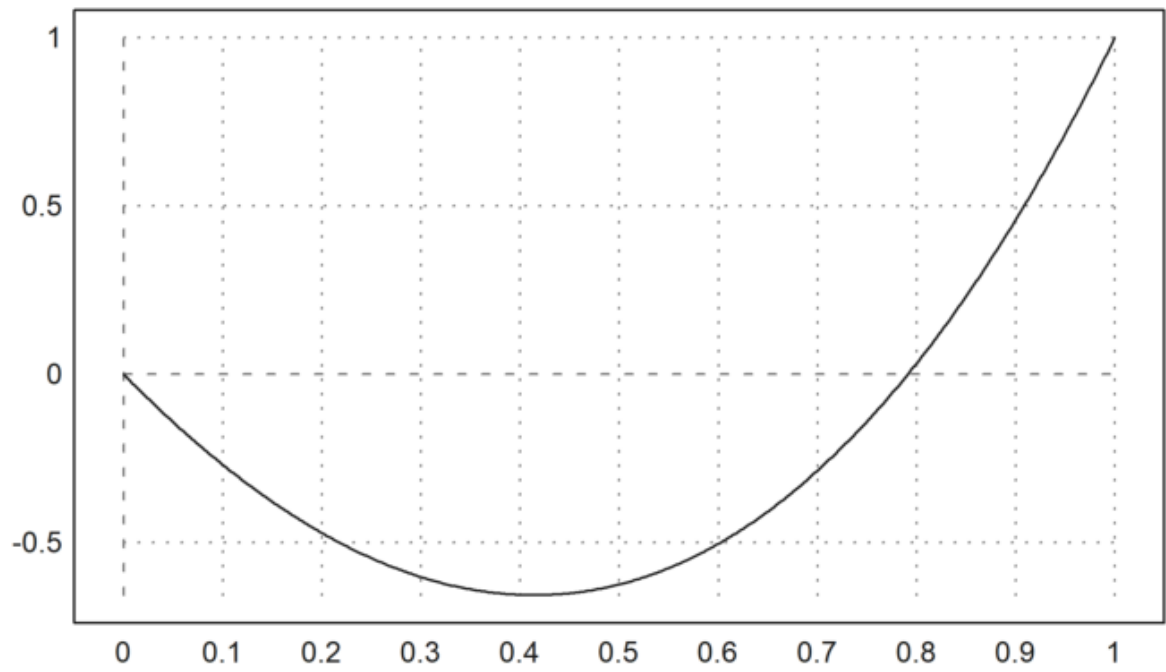
Fungsi dalam satu parameter adalah fungsi yang memerlukan tepat satu nilai masukan untuk dapat beroperasi.

Membuat plot fungsi:

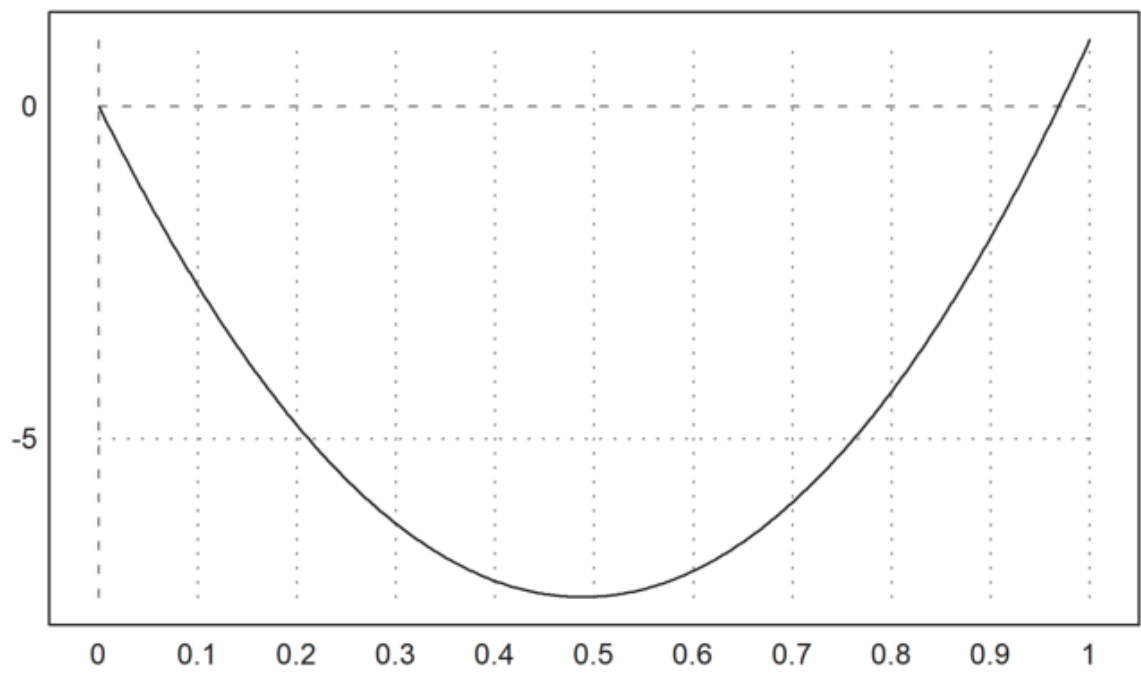
$$f(x, a) = x^3 + ax^2 - ax$$

dengan $a=3$

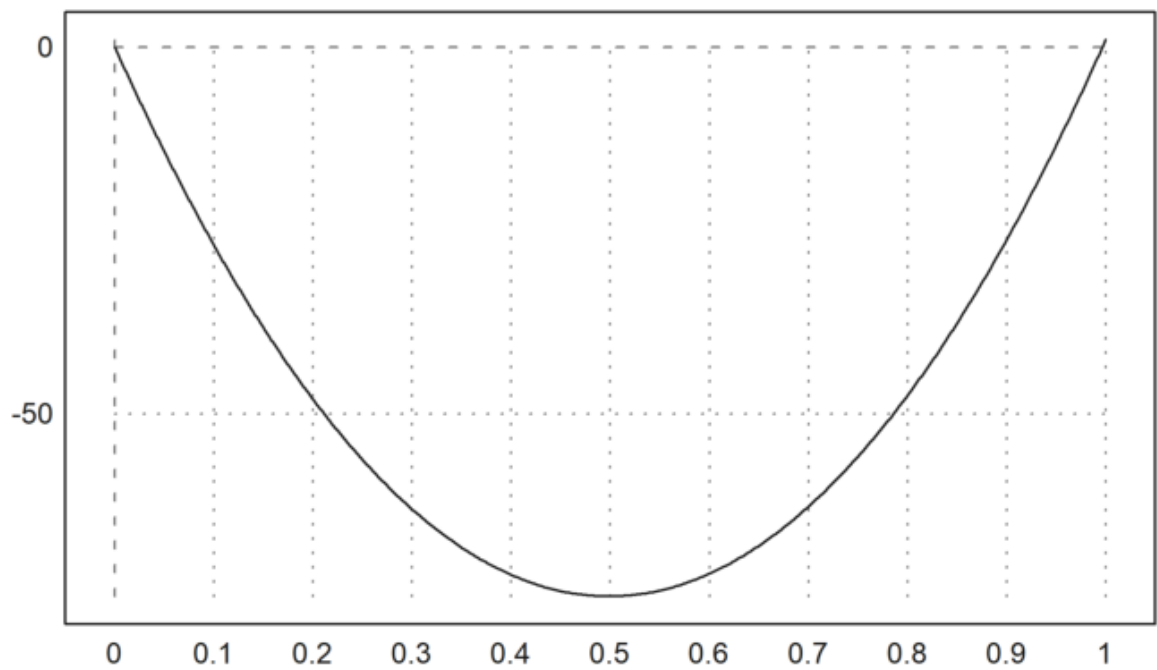
```
>function f(x,a) := x^3+a*x^2-a*x; //mendefinisikan fungsi  
>a=3; plot2d("f",0,1;a): //membuat plot dengan nilai a=3
```



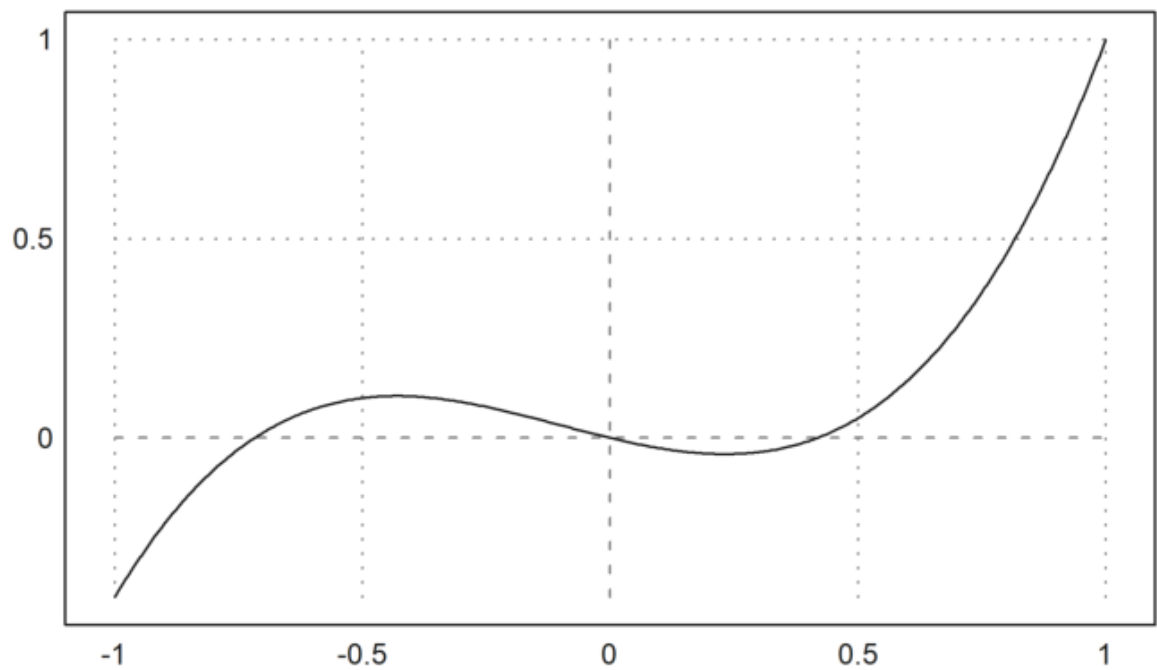
```
>a=30; plot2d("f",0,1;a): //membuat plot dengan nilai a=30
```



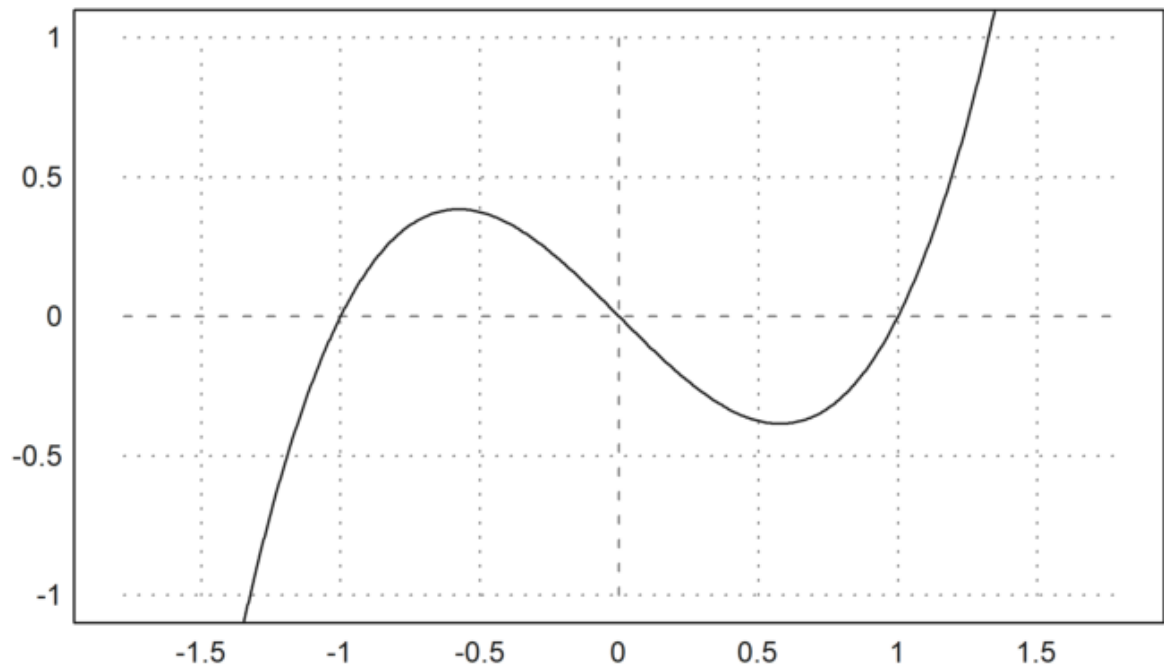
```
>a=300; plot2d("f",0,1;a): //membuat plot dengan nilai a=300
```



```
>a=0.3; plot2d("f",-1,1;a): //membuat plot dengan nilai a=0.3
```



```
>function f(x) := x^3-x;...
>plot2d("f",r=1): //r sebagai range dari -1 hingga 1
```



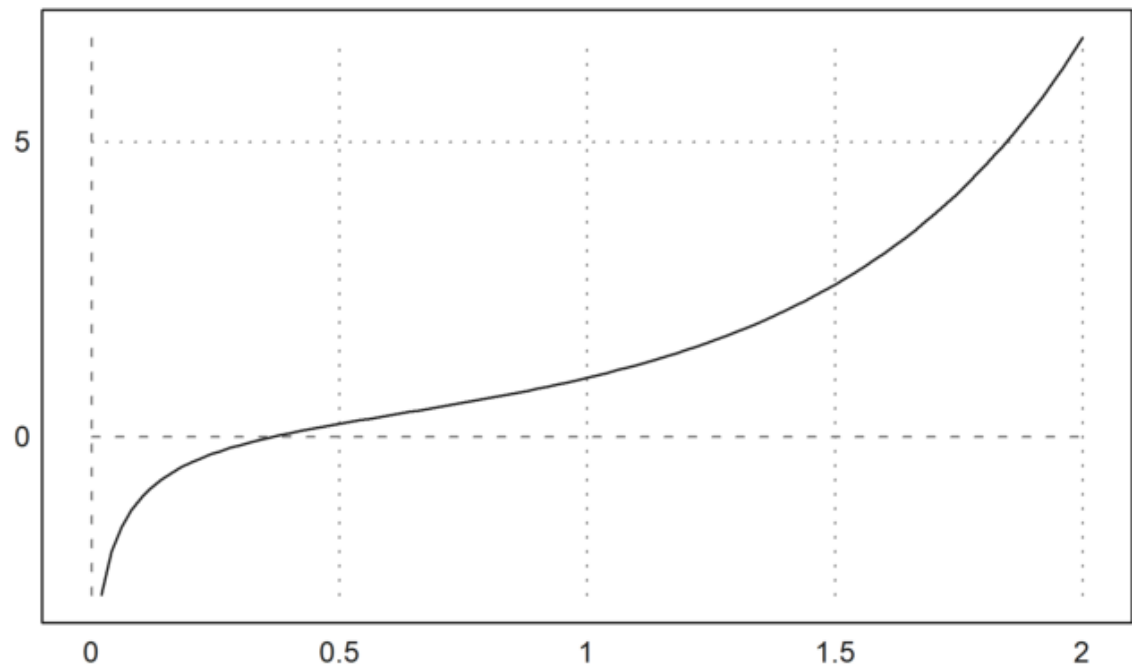
Berikut adalah ringkasan dari fungsi yang diterima

- ekspresi atau ekspresi simbolik dalam x, artinya Anda dapat ditulis ekspresi matematis
 - fungsi atau fungsi simbolis dengan nama sebagai "f"
 - fungsi simbolis hanya dengan nama f, jika sudah dinamai f maka selanjutnya mesti disebut f.
- Fungsi plot2d() juga menerima fungsi simbolis. Untuk fungsi simbolis, nama saja yang berfungsi.

```
>function f(x)&= diff(x^x,x) //fungsi g(x) adalah fungsi diferensial dari x^x terhadap x.
```

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

```
>plot2d(f,0,2): //membuat plot fungsi g(x) dari x=0 hingga x=2
```

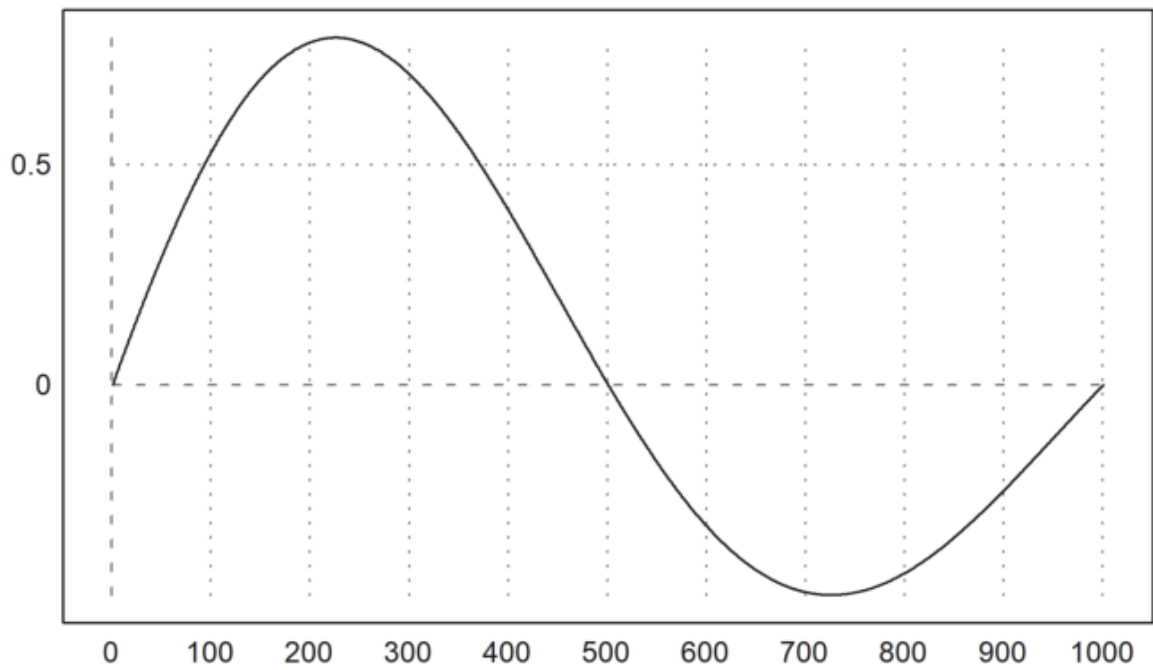


Tentu saja, untuk ekspresi atau ekspresi simbolik, nama variabel sudah cukup untuk memplotnya.

```
>y &= sin(x)*exp(-x) //y adalah nama, &= adalah operator definisi eksp simbolik
```

$$E^{-x} \sin(x)$$

```
>plot2d(y,0,3pi): //membuat plot fungsi y dari x=0 hingga x=3pi
```



Rahma Amalia (24030130028)

Label Teks

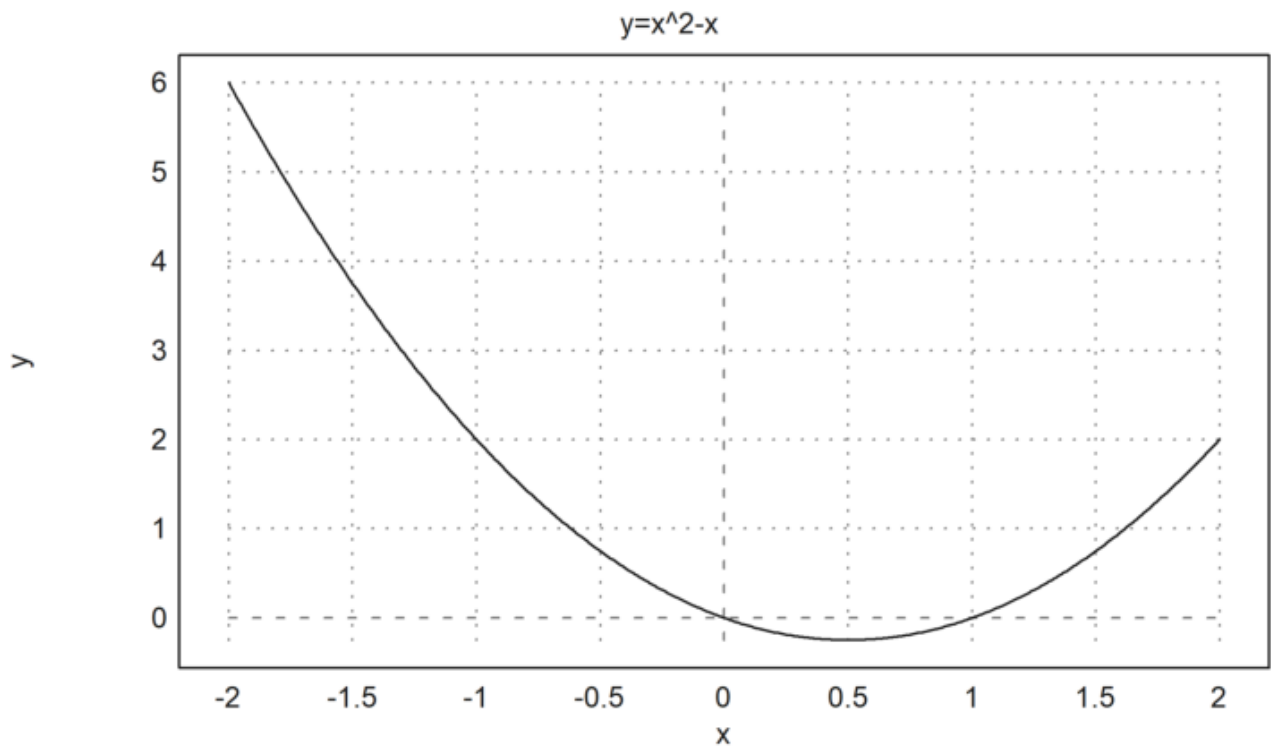
Menampilkan label teks dapat melakukan sebagai berikut:

- judul plot dengan title="..."
- label sumbu x dan sumbu y dengan xl="...", yl="..."
- label teks tambahan di posisi tertentu dengan label("...",x,y)

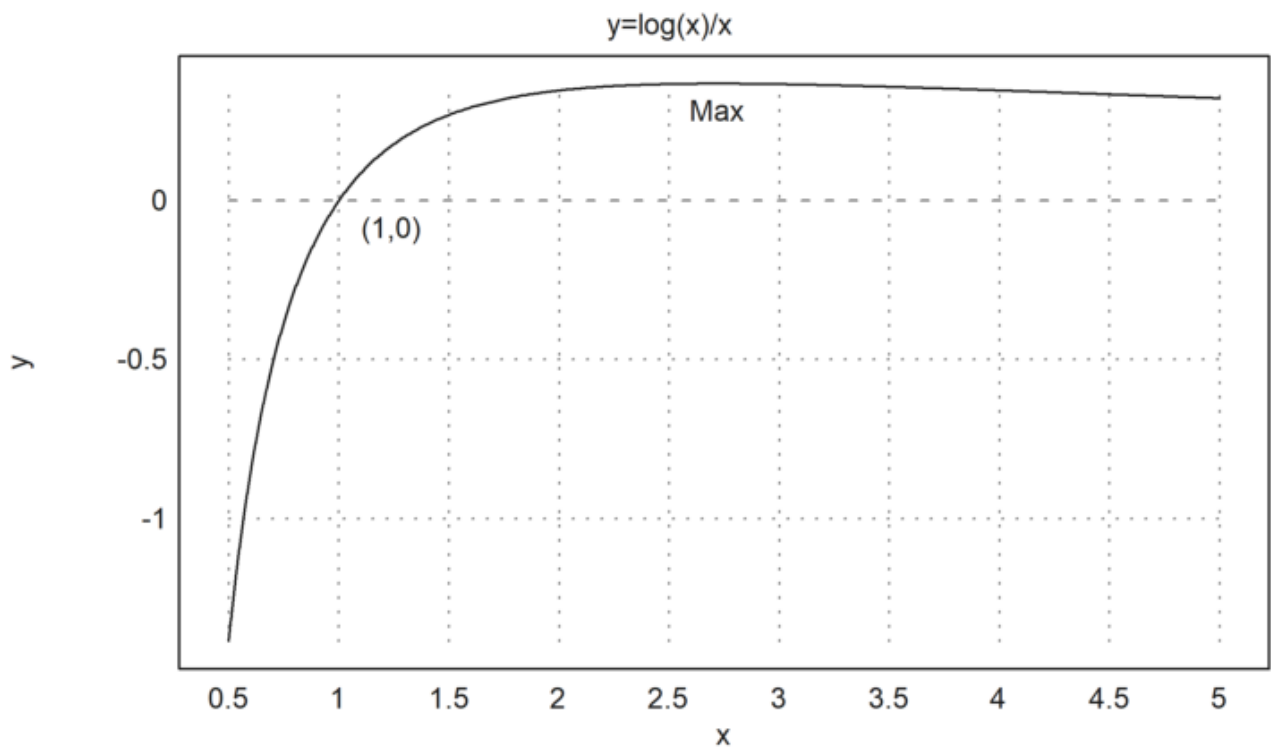
Perintah label akan memplot ke dalam plot saat ini pada koordinat plot (x,y).

- penulisan perintah untuk menampilkan tidak ada aturan urutan.

```
>plot2d("x^2-x",-2,2,title="y=x^2-x",yl="y",xl="x") :
```

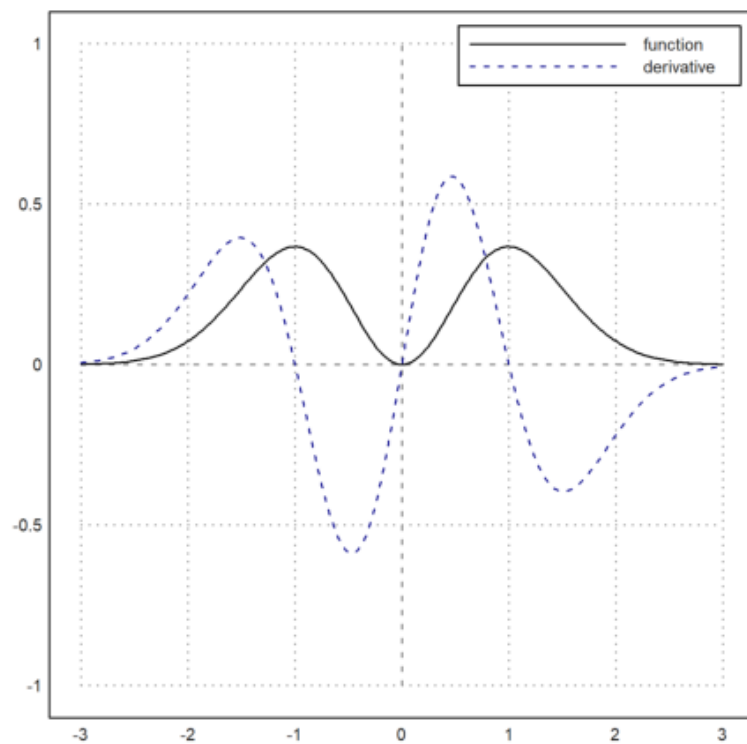


```
>expr := "log(x)/x"; //expr adalah ekspresi simbolik
>plot2d(expr,0.5,5,title="y="+expr,xl="x",yl="y"); //menggambar plot fungsi expr dari x=0.
>label("(1,0)",1,0); label("Max",E,expr(E),pos="lc"): //menampilkan (1,0), Max diatur di s
```



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

```
>function f(x) &= x^2*exp(-x^2); //mendefinisikan fungsi f(x)
>plot2d(&f(x),a=-3,b=3,c=-1,d=1); //menggambar plot fungsi f(x) x dari -3 hingga 3 dan y d
>plot2d(&diff(f(x),x),>add, color=blue, style="--"); //membuat plot fungsi diferensial f(x)
>labelbox(["function","derivative"],styles=["-", "--"],colors=[black,blue],w=0.4):
```

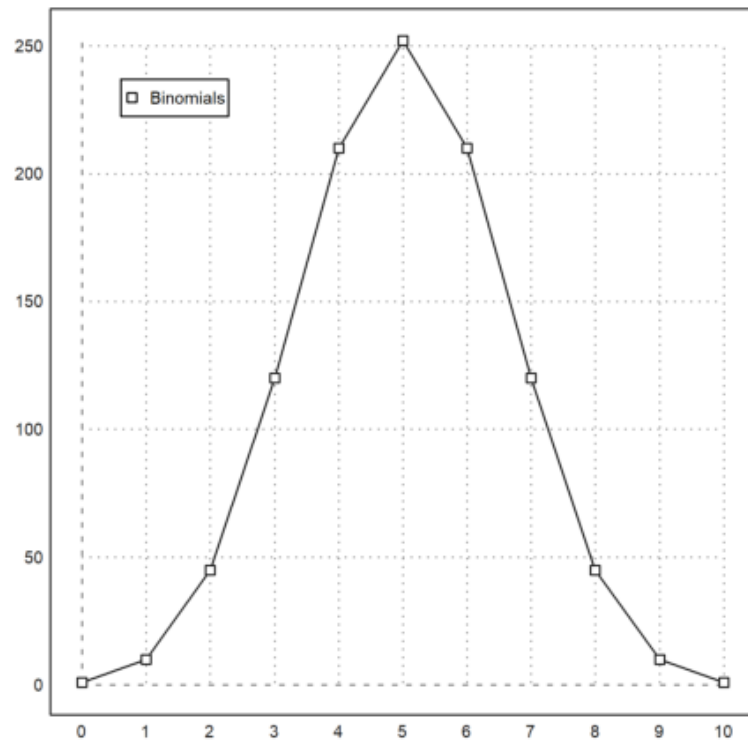


Kotak ditambatkan di kanan atas secara default, tetapi `>left` menambatkannya di kiri atas. Anda dapat memindahkannya ke tempat yang Anda suka. Posisi jangkar adalah sudut kanan atas kotak, dan angkanya adalah pecahan dari ukuran jendela grafik. Lebarannya otomatis.

Untuk plot titik, kotak label juga berfungsi. Tambahkan parameter `>points`, atau vektor flag, satu untuk setiap label.

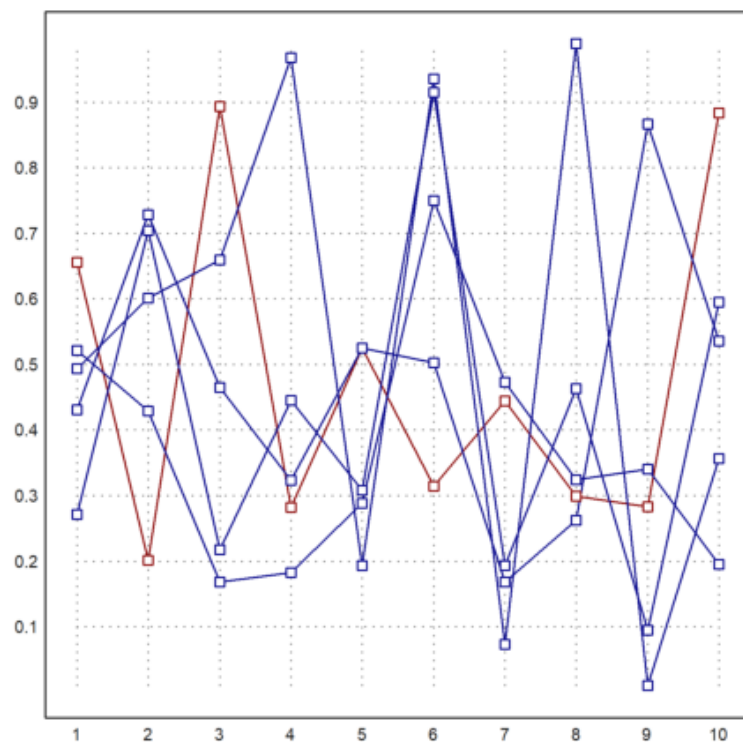
Dalam contoh berikut, hanya ada satu fungsi. Jadi kita bisa menggunakan string sebagai pengganti vektor string. Kami mengatur warna teks menjadi hitam untuk contoh ini.

```
>n=10; plot2d(0:n,bin(n,0:n),>addpoints); //menggambar distribusi binomial dengan n=10
>labelbox("Binomials",styles="[]",>points,x=0.1,y=0.1,tcolor=black,>left):
```

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

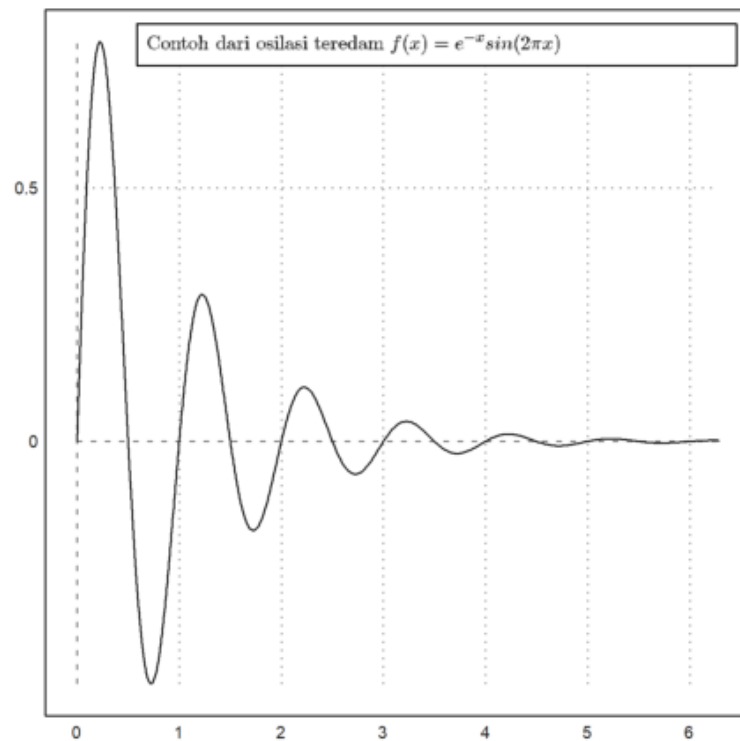
```
>statplot(1:10,random(5,10),color=[red,blue]):
```



Fitur serupa adalah fungsi `textbox()`.

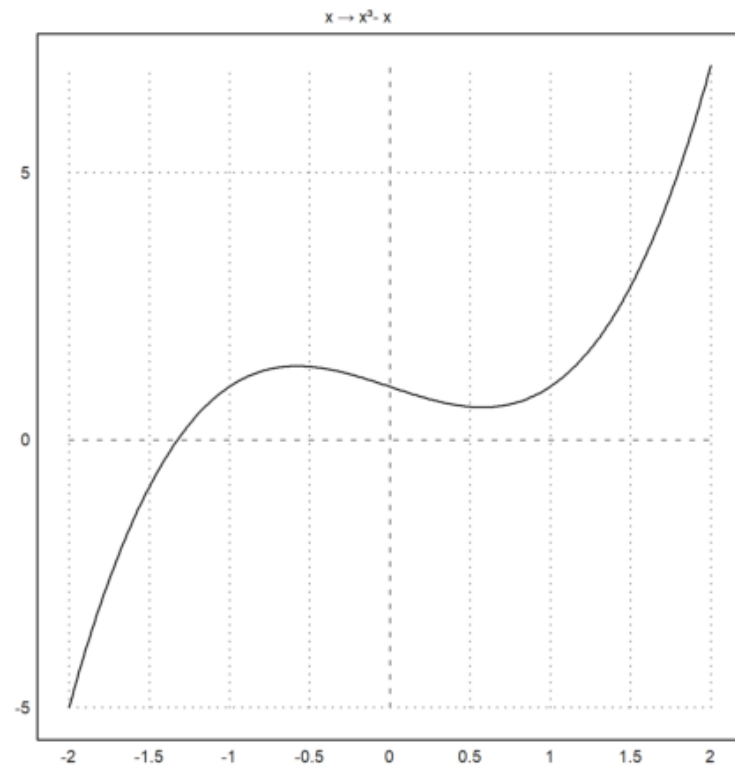
Lebar secara default adalah lebar maksimal dari baris teks. Tapi itu bisa diatur oleh pengguna juga.

```
>function f(x) &= exp(-x)*sin(2*pi*x); //mendefinisikan fungsi f(x)
>plot2d("f(x)",0,2pi); //menggambar plot f(x) dari x=0 hingga x=2pi
>textbox(latex("\text{Contoh dari osilasi teredam}\ f(x)=e^{-x}\sin(2\pi x)"),w=0.85):
```



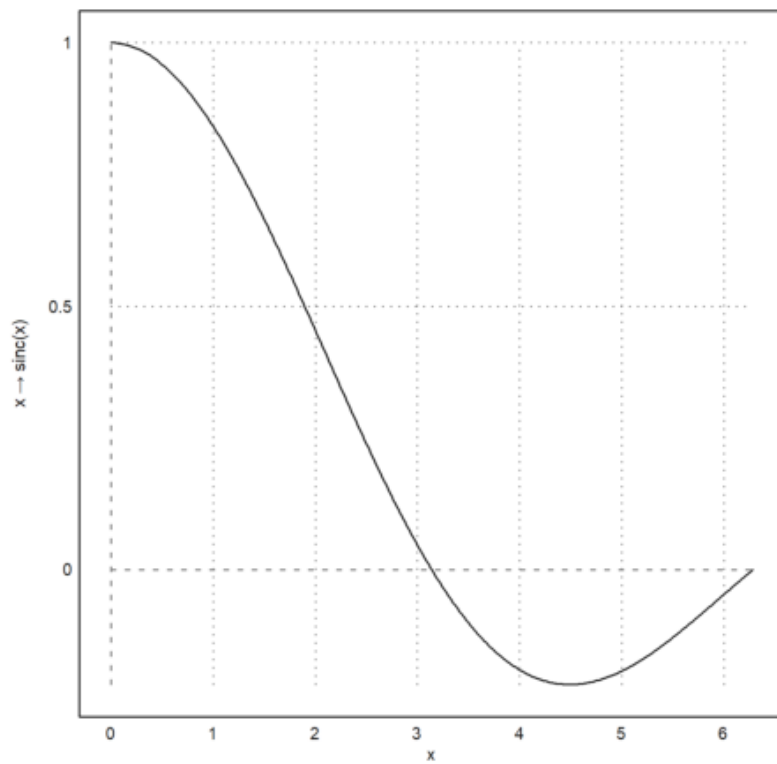
Label teks, judul, kotak label, dan teks lainnya dapat berisi string Unicode (lihat sintaks EMT untuk mengetahui lebih lanjut tentang string Unicode).

```
>plot2d("x^3-x+1",title=u"x \rarr; x&sup3;- x"): //&rarr adalah panah kanan
```



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

```
>plot2d("sinc(x)",0,2pi,xl="x",yl=u"x &rarr; sinc(x)",>vertical): //agar penulisan sinc(x)
```

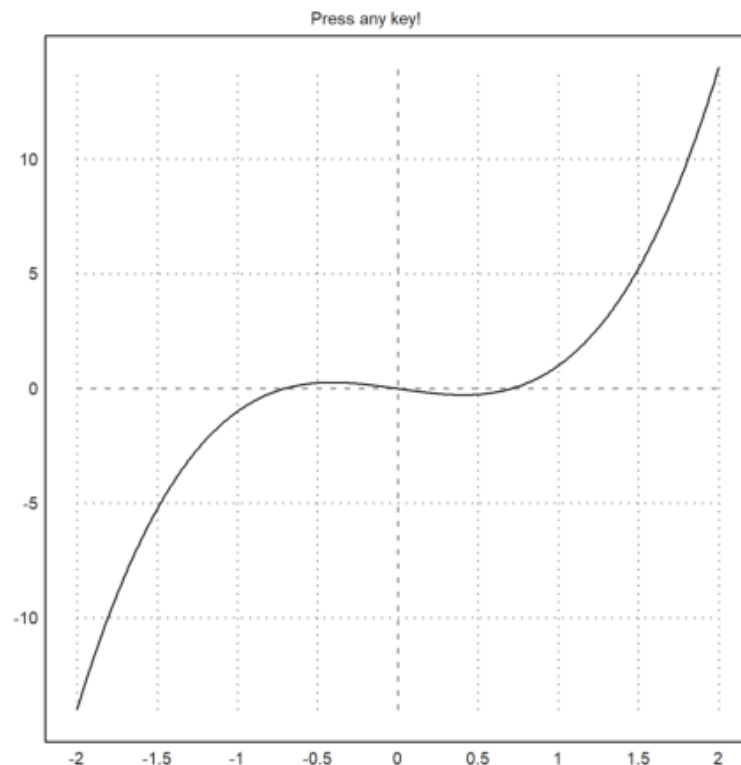


Saat memplot fungsi atau ekspresi, parameter `>user` memungkinkan pengguna untuk memperbesar dan menggeser plot dengan tombol kursor atau mouse. Pengguna dapat:

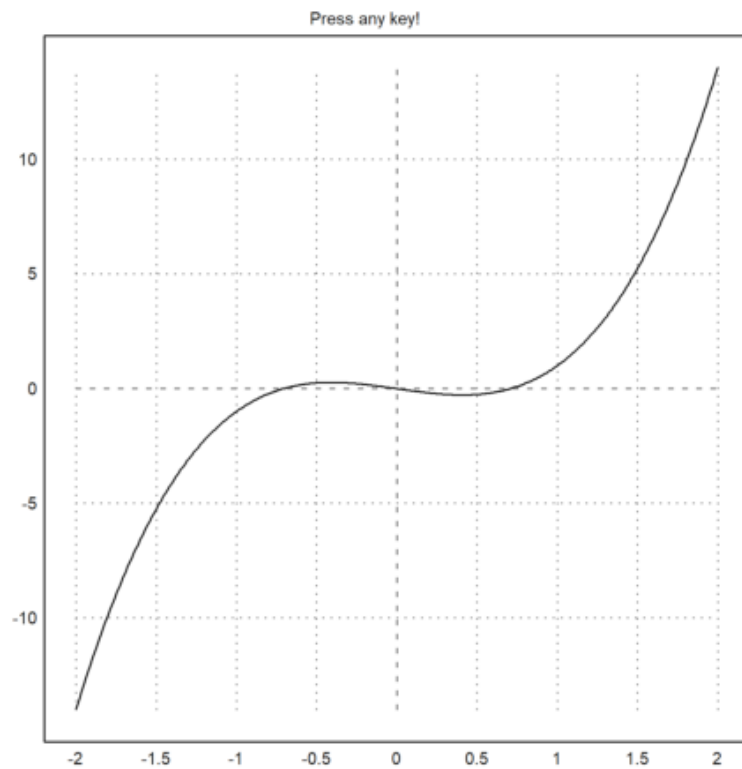
- memperbesar dengan + atau -
- memindah plot dengan tombol kursor
- memilih jendela plot dengan mouse
- mengatur ulang tampilan dengan spasi
- keluar dengan kembali

Tombol spasi akan mengatur ulang plot ke jendela plot asli. Saat memplot data, flag `>user` hanya akan menunggu penekanan tombol

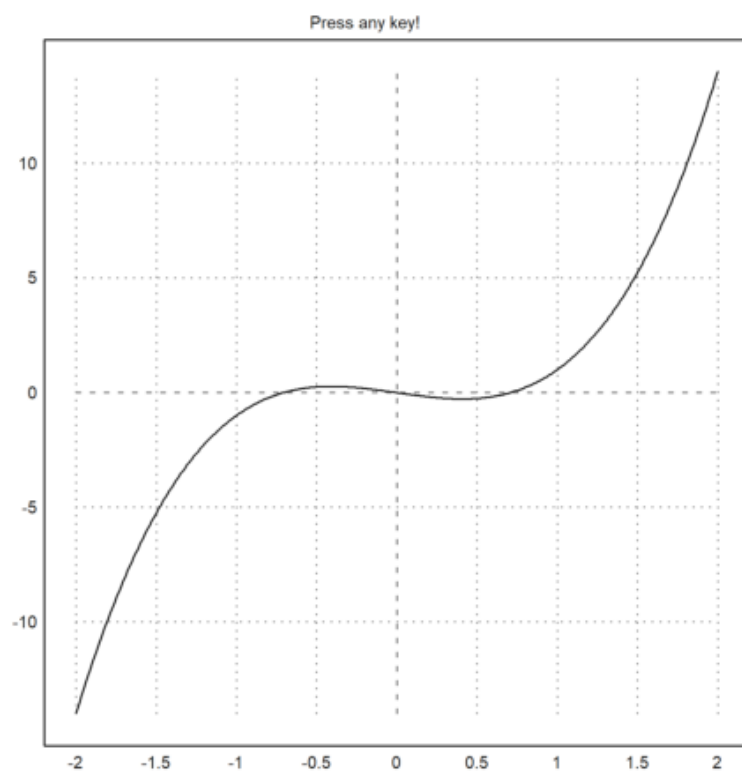
```
>plot2d({{"2*x^3-a*x",a=1}},>user,title="Press any key!"); ... //dengan fungsi yang sama
>insimg;
```



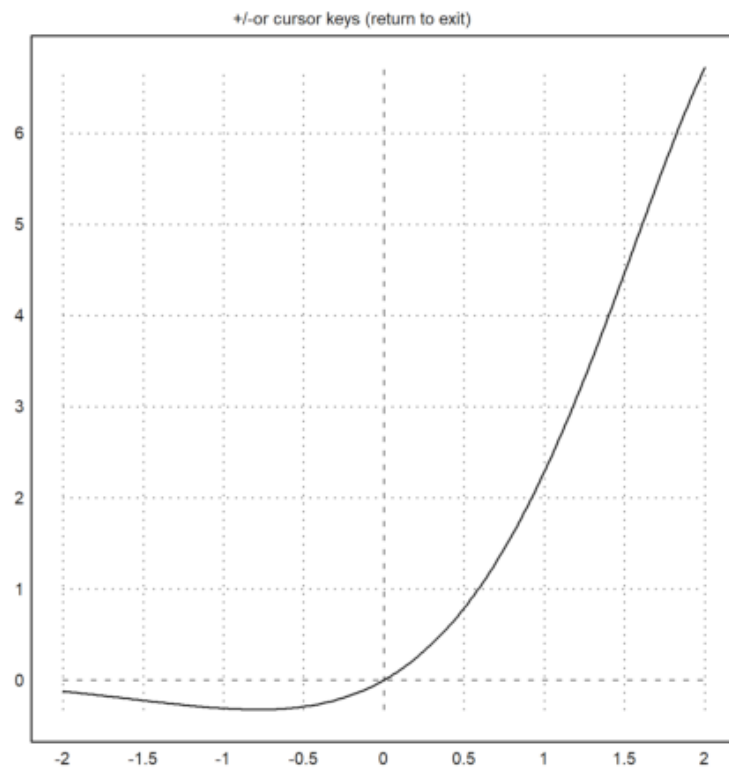
```
>plot2d({{"2*x^3-a*x",a=1}},>user,title="Press any key!"); ... //dengan fungsi yang sama
>insimg;
```



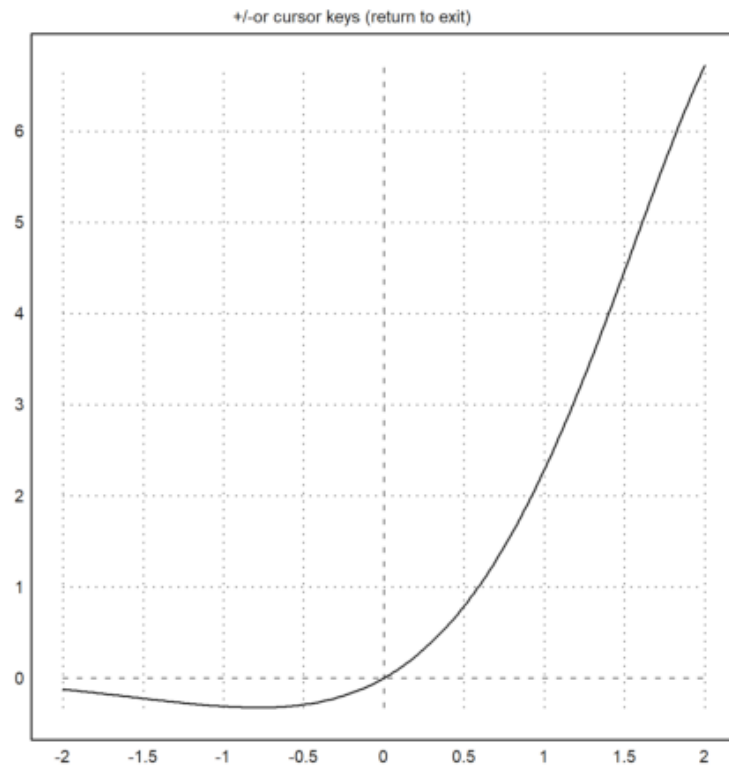
```
>plot2d({{"2*x^3-a*x",a=1}},>user,title="Press any key!"); ... //dengan fungsi yang sama  
>insimg;
```



```
>plot2d("exp(x)*sin(x)",user=true,... //user=true untuk mengaktifkan user interaction  
>title="+/-or cursor keys (return to exit)":
```



```
>plot2d("exp(x)*sin(x)",user=true,... //user=true untuk mengaktifkan user interaction  
>title="+/-or cursor keys (return to exit)":
```



>

Aji Handono Warih (21301241029)

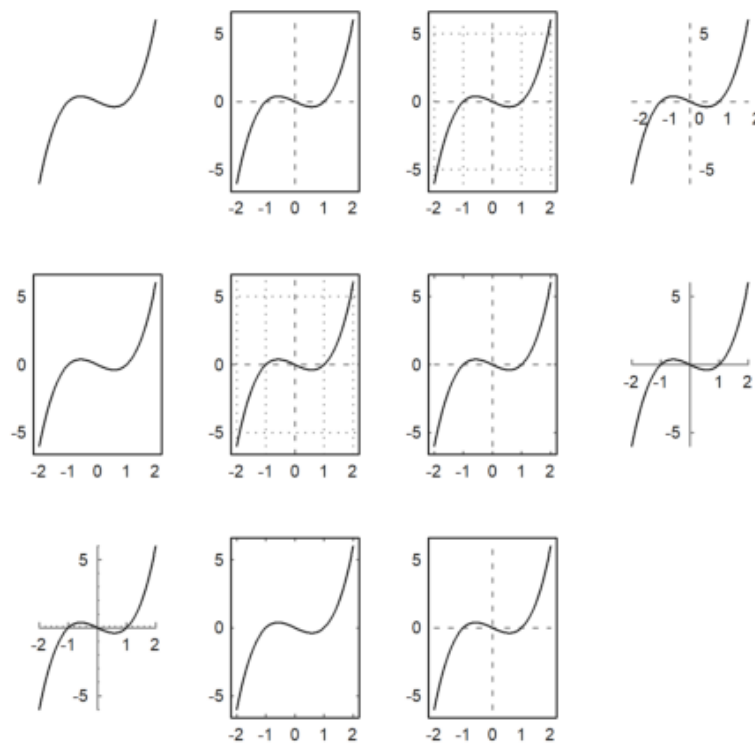
Format Plot 2D

Secara bawaan, EMT akan menghitung tanda sumbu (axis ticks) secara otomatis dan menambahkan label pada setiap tanda tersebut. Hal ini dapat diubah dengan menggunakan parameter grid. Gaya bawaan dari sumbu dan label juga dapat dimodifikasi. Selain itu, label dan judul dapat ditambahkan secara manual. Untuk mengembalikan ke gaya bawaan, gunakan fungsi reset().

```

>aspect();
>figure(3,4);
>figure(1); plot2d("x^3-x",grid=0); // no grid, frame or axis
>figure(2); plot2d("x^3-x",grid=1); // x-y-axis
>figure(3); plot2d("x^3-x",grid=2); // default ticks
>figure(4); plot2d("x^3-x",grid=3); // x-y- axis with labels inside
>figure(5); plot2d("x^3-x",grid=4); // no ticks, only labels
>figure(6); plot2d("x^3-x",grid=5); // default, but no margin
>figure(7); plot2d("x^3-x",grid=6); // axes only
>figure(8); plot2d("x^3-x",grid=7); // axes only, ticks at axis
>figure(9); plot2d("x^3-x",grid=8); // axes only, finer ticks at axis
>figure(10); plot2d("x^3-x",grid=9); // default, small ticks inside
>figure(11); plot2d("x^3-x",grid=10); // no ticks, axes only
>figure(0):

```



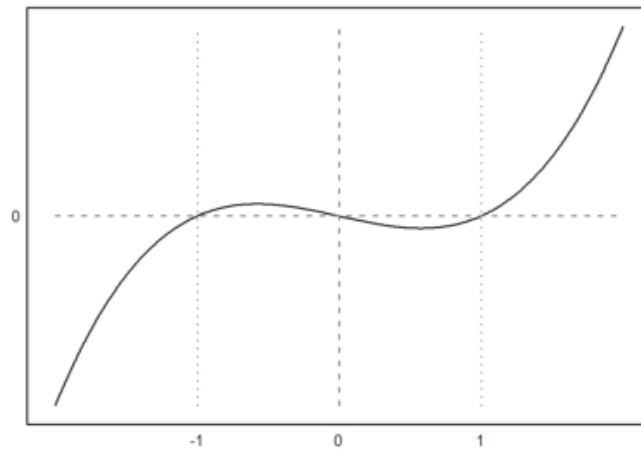
Parameter <frame akan menonaktifkan bingkai (frame), sedangkan framecolor=blue akan mengatur bingkai menjadi berwarna biru.

Jika Anda ingin membuat tanda sumbu (ticks) sendiri, Anda dapat menggunakan grid=0, lalu menambahkan semua elemen secara manual setelahnya.

```

>aspect(1.5);
>plot2d("x^3-x",grid=0); // plot
>frame; xgrid([-1,0,1]); ygrid(0): // menambahkan frame dan grid

```

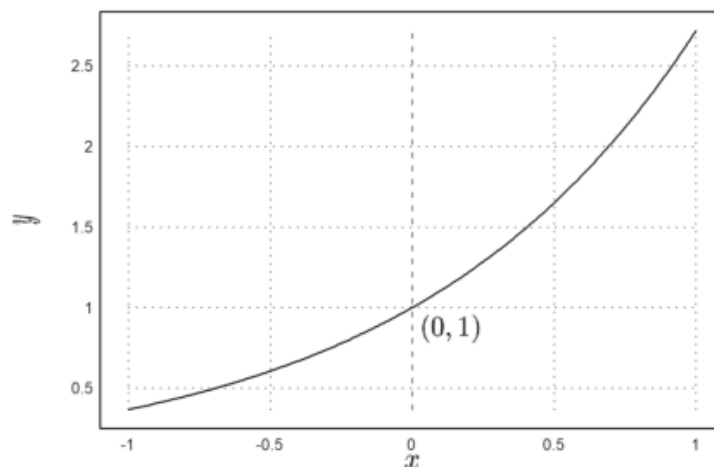



Untuk judul plot dan label pada sumbu, perhatikan contoh berikut.

```
>plot2d("exp(x)",-1,1);
>plot2d("exp(x)",-1,1);
>textcolor(black); // set the text color to black
>title (latex("y=e^x")); // title above the plot
```

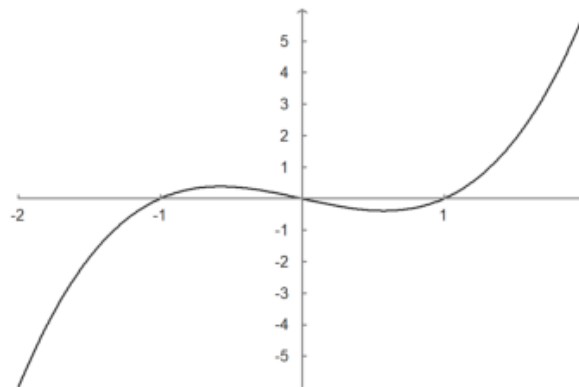
```
Syntax error in expression, or unfinished expression!
Error in expression: +/-or cursor keys (return to exit)
Error in:
title (latex("y=e^x")); // title above the plot ...
      ^
```

```
>xlabel(latex("x")); // "x" for x-axis
>ylabel(latex("y"),>vertical); // vertical "y" for y-axis
>label(latex("(0,1)"),0,1,color=blue): // label a point
```



Sumbu dapat digambar secara terpisah dengan menggunakan `xaxis()` dan `yaxis()`

```
>plot2d("x^3-x",<grid,<frame);
>xaxis(0,xx=-2:1,style="->"); yaxis(0,yy=-5:5,style="->"):
```

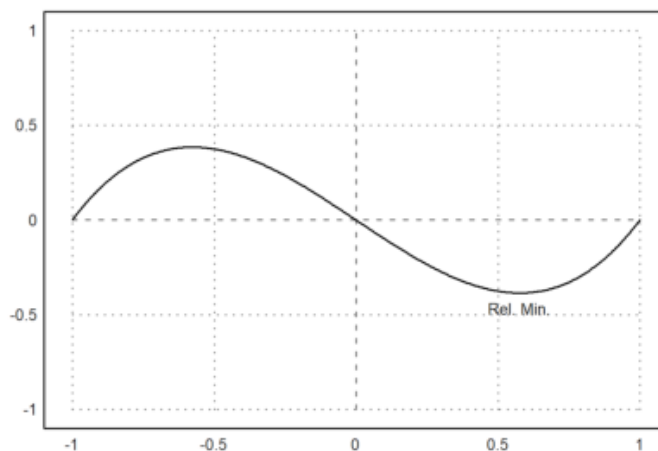


Teks pada plot dapat diatur dengan menggunakan label(). Pada contoh berikut, "lc" berarti lower center (tengah bawah). Kode ini digunakan untuk mengatur posisi label relatif terhadap koordinat plot.

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

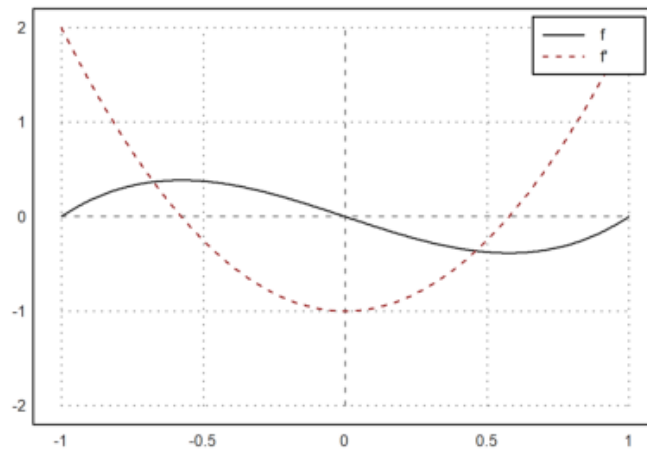
$$x^3 - x$$

```
>plot2d(f,-1,1,>square);
>x0=fmin(f,0,1); // compute point of minimum
>label("Rel. Min.",x0,f(x0),pos="lc"): // add a label there
```

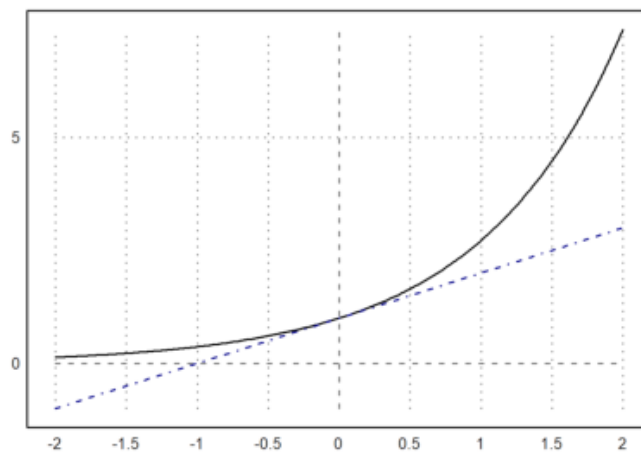


Terdapat juga kotak teks (text boxes).

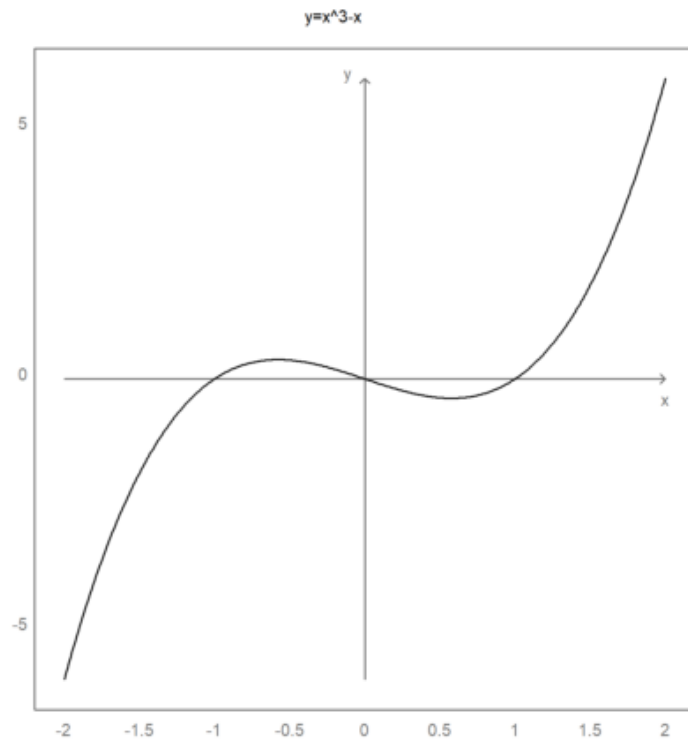
```
>plot2d(&f(x),-1,1,-2,2); // function
>plot2d(&diff(f(x),x),>add,style="--",color=red); // derivative
>labelbox(["f","f'"],["-","--"],[black,red]): // label box
```



```
>plot2d(["exp(x)", "1+x"],color=[black,blue],style=["-","-.-"]):
```



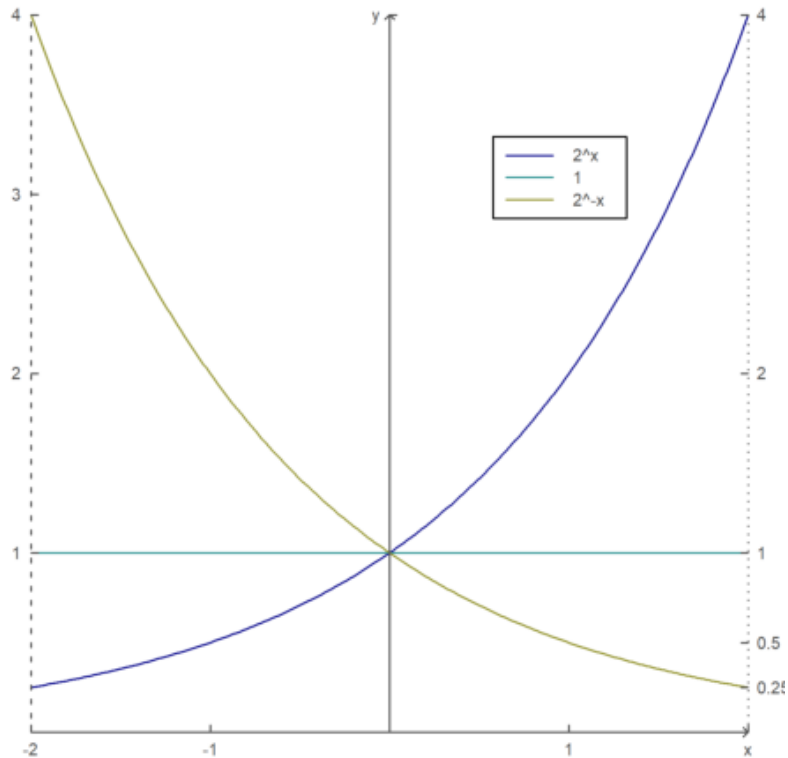
```
>gridstyle("->",color=gray,textcolor=gray,framecolor=gray);
> plot2d("x^3-x",grid=1);
> setttitle("y=x^3-x",color=black);
> label("x",2,0,pos="bc",color=gray);
> label("y",0,6,pos="cl",color=gray);
> reset():
```



Untuk kendali yang lebih lanjut, sumbu-x dan sumbu-y dapat dibuat secara manual.

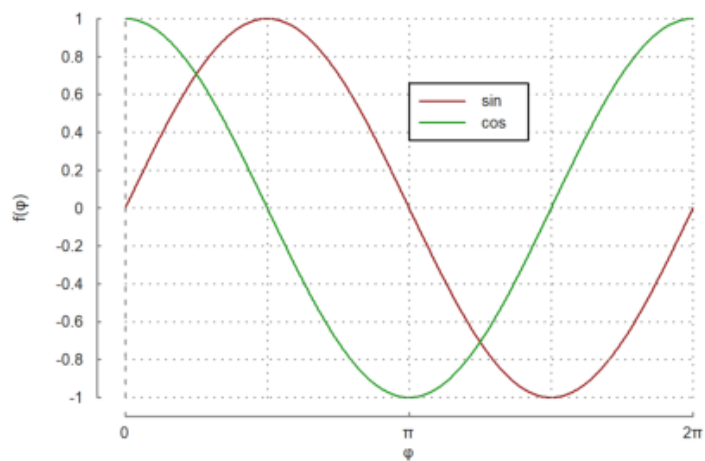
Perintah `fullwindow()` akan memperluas jendela plot karena kita tidak lagi memerlukan ruang untuk label di luar jendela plot. Gunakan `shrinkwindow()` atau `reset()` untuk mengembalikannya ke pengaturan bawaan.

```
>fullwindow;
> gridstyle(color=darkgray,textcolor=darkgray);
> plot2d(["2^x","1","2^(-x)"],a=-2,b=2,c=0,d=4,<grid,color=4:6,<frame);
> xaxis(0,-2:1,style="->"); xaxis(0,2,"x",<axis);
> yaxis(0,4,"y",style="->");
> yaxis(-2,1:4,>left);
> yaxis(2,2^(-2:2),style=".",<left);
> labelbox(["2^x","1","2^-x"],colors=4:6,x=0.8,y=0.2);
> reset;
```



Berikut contoh lain, di mana digunakan string Unicode dan sumbu ditempatkan 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"&pi;", u"2&pi;"], 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"&phi;"); ylabel(u"f(&phi;)"):
```

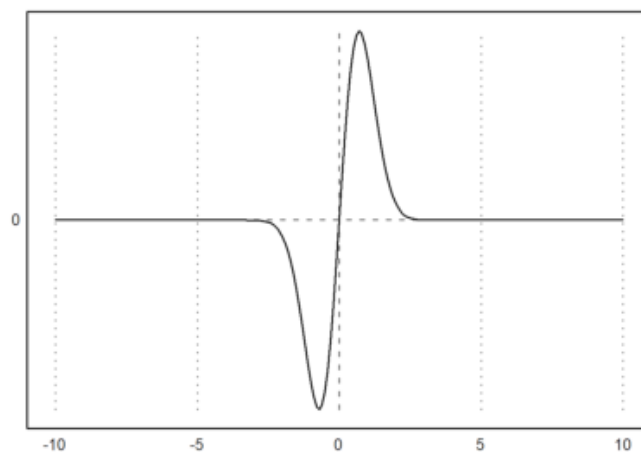


Aji Handono Warih (21301241029)

Memplot sebuah ekspresi hanyalah bentuk singkat dari pemetaan data. Untuk pemetaan data, Anda memerlukan satu atau lebih baris nilai x, dan satu atau lebih baris nilai y. Dari rentang dan nilai x tersebut, fungsi `plot2d` akan menghitung data yang akan diplot, secara bawaan dengan evaluasi adaptif dari fungsi. Untuk plot titik gunakan `>points`, Untuk campuran garis dan titik gunakan `>addpoints`.

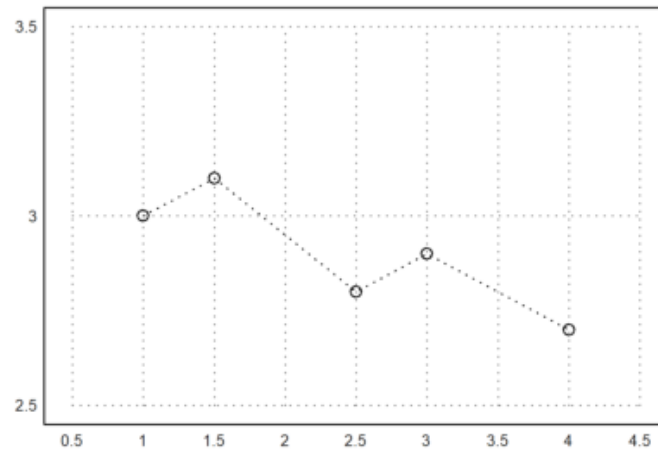
- Gunakan vektor baris untuk x dan y pada satu fungsi.
- Matriks untuk x dan y akan diplot baris demi baris.

```
>x=-10:0.1:10; y=exp(-x^2)*x; plot2d(x,y):
```

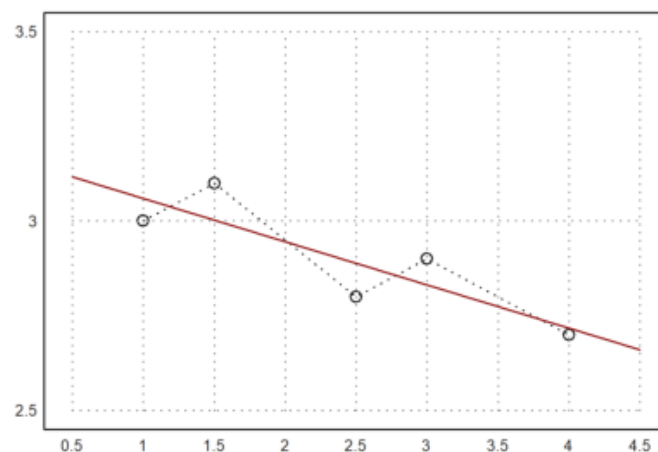


- style="...": Pilih dari "[", "<>", "o", ".", "..", "+", "*", "[", "<>", "o", ".", "", "|".

```
>xdata=[1,1.5,2.5,3,4]; ydata=[3,3.1,2.8,2.9,2.7]; // data
>plot2d(xdata,ydata,a=0.5,b=4.5,c=2.5,d=3.5,style="."); // lines
>plot2d(xdata,ydata,>points,>add,style="o"): // add points
```



```
>p=polyfit(xdata,ydata,1); // get regression line
>plot2d("polyval(p,x)",>add,color=red): // add plot of line
```



Aji Handono Warih (21301241029)

Menggambar Daerah Yang Dibatasi Kurva

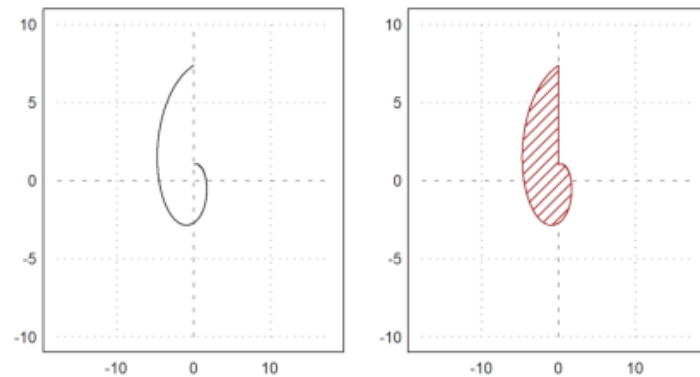
Plot data pada dasarnya adalah poligon. Kita juga dapat memplot kurva atau kurva terisi (filled curves).

- filled=true akan mengisi (fill) area pada plot.
- style="...": dipilih dari "", "/", "\", "\/".
- fillcolor: melihat daftar warna yang tersedia pada penjelasan di atas.

Warna isian ditentukan oleh argumen "fillcolor", dan opsi <outline bersifat opsional untuk mencegah digambarkan batas (boundary) pada semua gaya kecuali gaya bawaan.

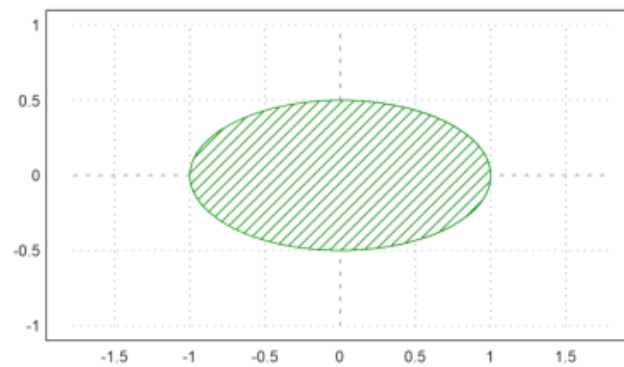
```
>t=linspace(0,2pi,1000); // parameter for curve
>x=sin(t)*exp(t/pi); y=cos(t)*exp(t/pi); // x(t) and y(t)
>figure(1,2); aspect(16/9) ;
```

```
>figure(1); plot2d(x,y,r=10); // plot curve
>figure(2); plot2d(x,y,r=10,>filled,style="/",fillcolor=red); // fill curve
>figure(0):
```

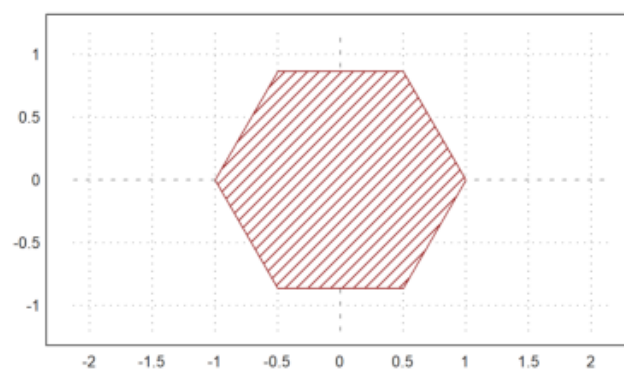


Pada contoh berikut, kita memplot sebuah elips terisi dan dua heksagon terisi dengan menggunakan kurva tertutup yang memiliki 6 titik, masing-masing dengan gaya isian (fill style) yang berbeda.

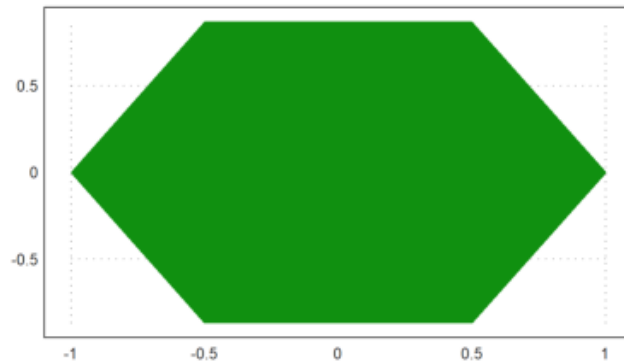
```
>x=linspace(0,2pi,1000); plot2d(sin(x),cos(x)*0.5,r=1,>filled,style="/"):
```



```
>t=linspace(0,2pi,6);
>plot2d(cos(t),sin(t),>filled,style="/",fillcolor=red,r=1.2):
```




```
>t=linspace(0,2pi,6); plot2d(cos(t),sin(t),>filled,style="#"):
```

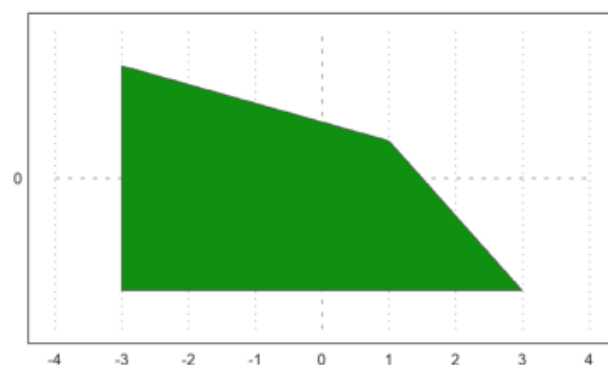


Contoh lainnya adalah septagon, yang dibuat dengan 7 titik pada lingkaran satuan (unit circle).

```
>t=linspace(0,2pi,7); ...  
> plot2d(cos(t),sin(t),r=1,>filled,style="/",fillcolor=red)
```

Berikut adalah himpunan dari nilai maksimum empat kondisi linear yang kurang dari atau sama dengan 3. Bentuknya adalah $A[k].v = 3$ untuk semua baris matriks A. Untuk mendapatkan sudut yang lebih halus, kita menggunakan nilai 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):
```

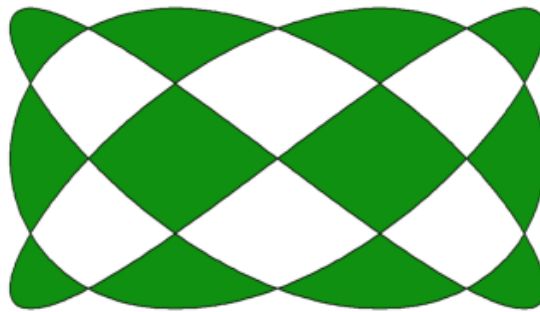


Inti utama dari bahasa matriks adalah kemampuannya untuk menghasilkan tabel fungsi dengan mudah.

```
>t=linspace(0,2pi,1000); x=cos(3*t); y=sin(4*t);
```

Sekarang kita memiliki vektor nilai x dan y . Fungsi `plot2d()` dapat memplot nilai-nilai ini sebagai kurva yang menghubungkan titik-titiknya. Plot tersebut juga dapat diisi (filled). Dalam kasus ini, hasilnya terlihat bagus karena menggunakan aturan winding (winding rule) untuk proses pengisian.

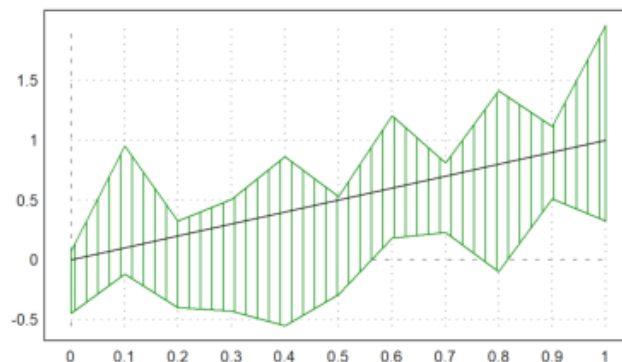
```
>plot2d(x,y,<grid,<frame,>filled):
```



Sebuah vektor interval diplot terhadap nilai x sebagai daerah terisi di antara nilai batas bawah dan batas atas interval tersebut.

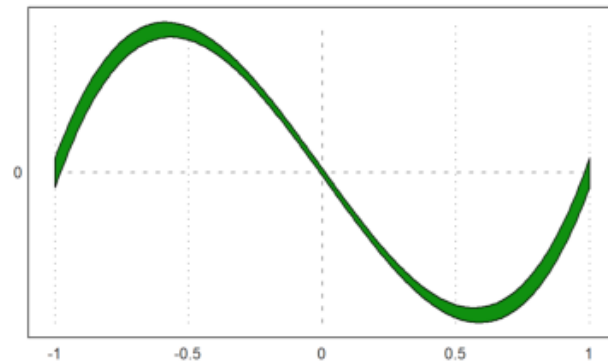
Hal ini dapat berguna untuk memplot galat perhitungan (computation errors), tetapi juga bisa digunakan untuk memvisualisasikan galat statistik (statistical errors).

```
>t=0:0.1:1;
> plot2d(t,interval(t-random(size(t)),t+random(size(t))),style="|");
> plot2d(t,t,add=true):
```



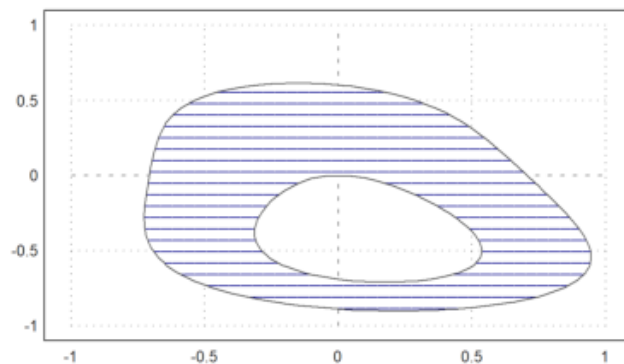
Jika x adalah sebuah vektor yang terurut, dan y adalah vektor interval, maka `plot2d` akan memplot rentang terisi dari interval tersebut pada bidang. Gaya isian (fill styles) yang digunakan sama dengan gaya pada poligon.

```
>t=-1:0.01:1; x=~t-0.01,t+0.01~; y=x^3-x;
>plot2d(t,y):
```



Dimungkinkan untuk mengisi (fill) daerah nilai dari suatu fungsi tertentu. Untuk keperluan ini, level harus berupa matriks berukuran $2 \times n$. Baris pertama berisi batas bawah, sedangkan baris kedua berisi batas atas.

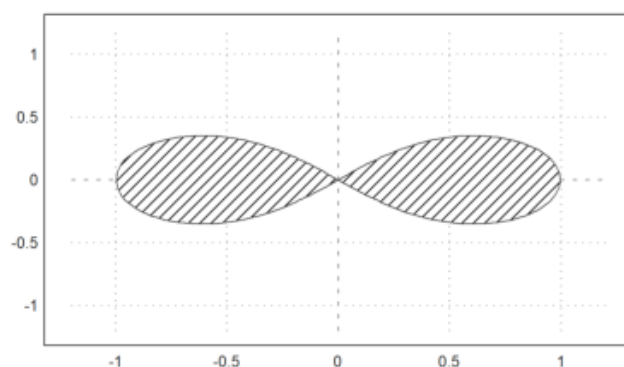
```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <= 1
```



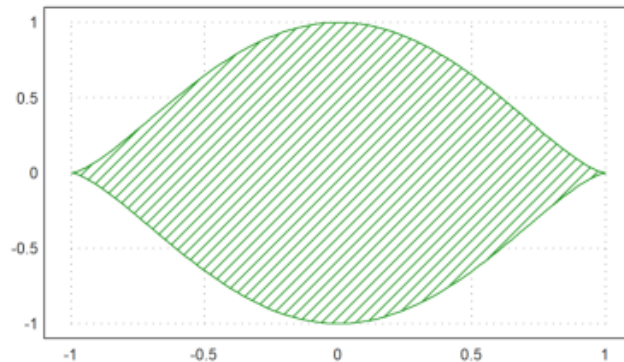
Kita juga dapat mengisi (fill) rentang nilai seperti

$$-1 \leq (x^2 + y^2)^2 - x^2 + y^2 \leq 0.$$

```
>plot2d("(x^2+y^2)^2-x^2+y^2",r=1.2,level=[-1;0],style="/"): 
```



```
>plot2d("cos(x)", "sin(x)^3", xmin=0, xmax=2pi, >filled, style="/") :
```



Princess Velaninta Audrey Sava (24030130075)

Fungsi Parametrik

Nilai-x tidak perlu diurutkan. (x,y) hanya menggambarkan kurva. Jika x diurutkan, kurva tersebut merupakan grafik fungsi.

Dalam contoh berikut, kami memplot spiral

$$\gamma(t) = t \cdot (\cos(s\pi t), \sin(2\pi t))$$

Kita perlu menggunakan banyak titik untuk tampilan yang halus atau fungsi adaptif() untuk mengevaluasi ekspresi (lihat fungsi) adaptif() untuk lebih jelasnya

```
>t=linspace(0,1,1000);
>plot2d(t*cos(2*pi*t), (t*sin(2*pi*t)), r=1);
```

Atau, dimungkinkan untuk menggunakan dua ekspresi untuk kurva

```
>plot2d("x*cos(2*pi*x)", "x*sin(2*pi*x)", xmin=0, xmax=1, r=1)
>t=linspace(0,1,1000); r=exp(-t); x=r*cos(2*pi*t); y=r*sin(2*pi*t);
>plot2d(x,y, r=1);
```

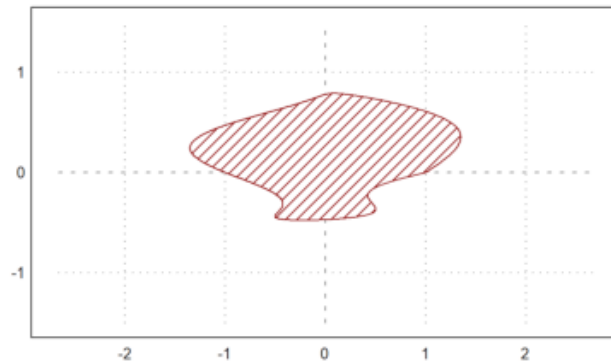
Dalam contoh berikutnya, kami memplot kurva

lateks: $\gamma(t) = (r(t)\cos(t), r(t)\sin(t))$

dengan

lateks: $r(t) = 1 + \frac{\sin(3t)}{2}$.

```
>t=linspace(0,2pi,1000); r=1+sin(3*t)/2; x=r*cos(t);
>plot2d(x,y, >filled, fillcolor=red, style="/", r=1.5) :
```

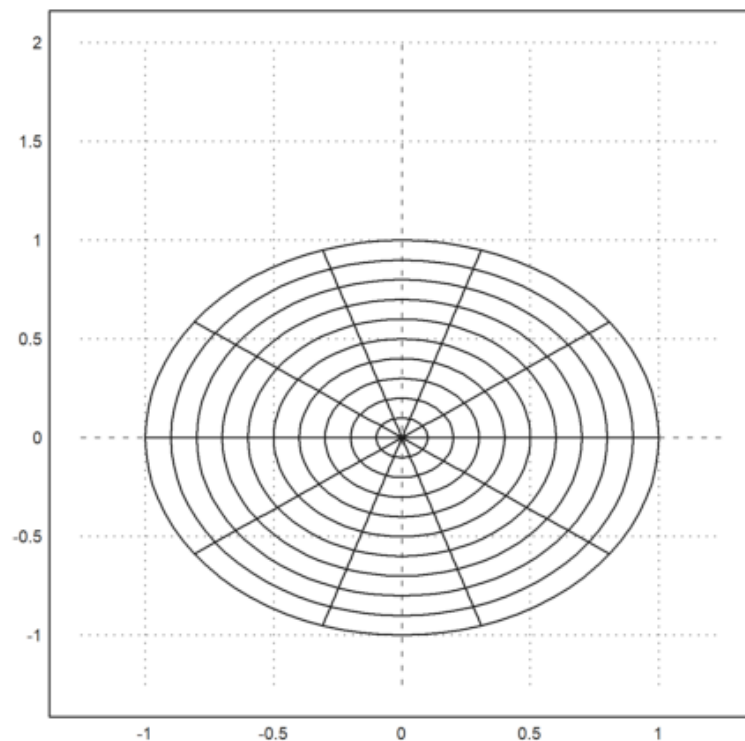


Princess Velaninta Audrey Sava (24030130075)

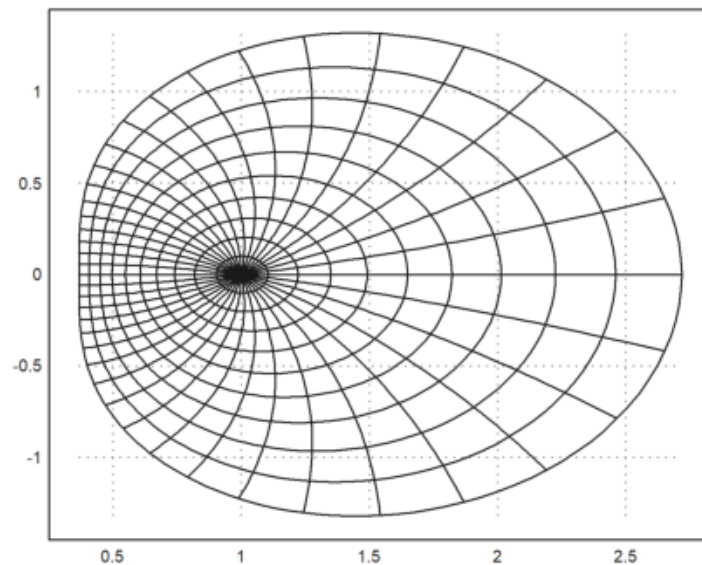
Grafik Bilangan Kompleks

Array bilangan kompleks juga dapat diplot yang kemudian titik-titik grid akan terhubung. Jika sejumlah garis kisi ditentukan (atau vektor garis kisi 1×2) dalam argumen `cgrid`, hanya garis kisi tersebut yang terlihat. Matriks bilangan kompleks akan secara otomatis diplot sebagai kisi di bidang kompleks. Dalam contoh berikut, kami memplot gambar lingkaran satuan di bawah fungsi eksponensial. Parameter `cgrid` menyembunyikan beberapa kurva grid.

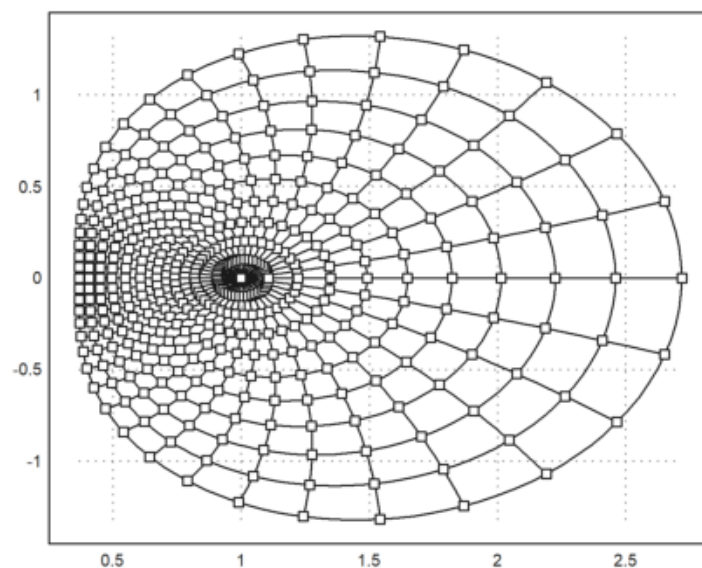
```
>aspect(); r=linspace(0,1,50); a=linspace(0,2pi,80)'; z=r*exp(I*a);
>plot2d(z,a=-1.25,b=1.25,c=-1.25,cgrid=10):
```



```
>aspect(1.25); r=linspace(0,1,50); a=linspace(0,2pi,200)'; z=r*exp(I*a);
>plot2d(exp(z),cgrid=[40,10]):
```

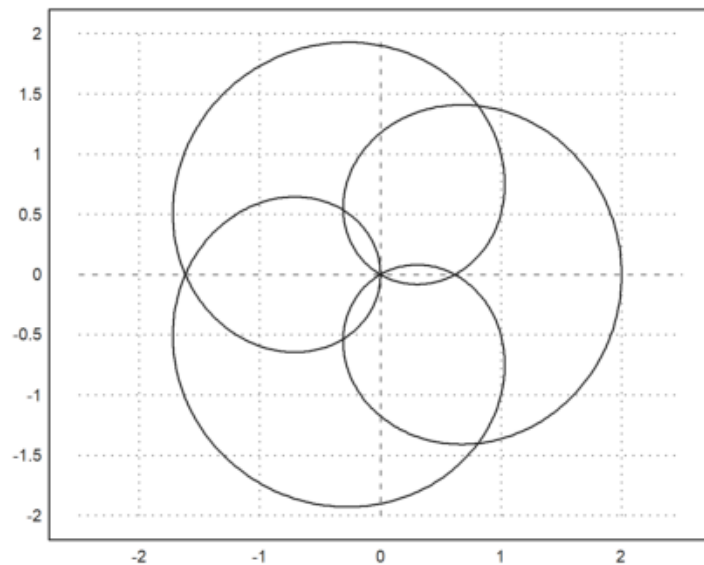


```
>r=linspace(0,1,10); a=linspace(0,2pi,40)'; z=r*exp(I*a);
>plot2d(exp(z),>points,>add):
```



Sebuah vektor bilangan kompleks secara otomatis diplot sebagai kurva pada bidang kompleks dengan bagian imajiner. Dalam contoh, kami memplot lingkaran satuan dengan
 $\gamma(t) = e^{it}$

```
>t=linspace(0,2pi,1000);
>plot2d(exp(I*t)+exp(4*I*t),r=2):
```



Princess Velaninta Audrey Sava (24030130075)

Fungsi Implisit

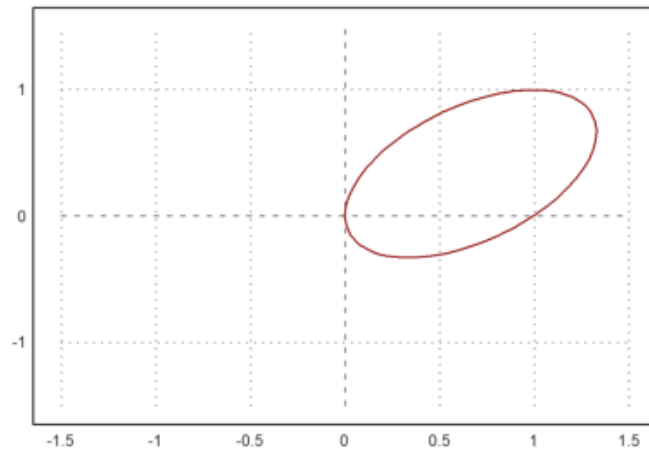
Plot implisit menunjukkan garis level yang menyelesaikan $f(x,y)=\text{level}$, di mana "level" dapat berupa nilai tunggal atau vektor nilai. Jika $\text{level}=\text{"auto"}$, akan ada garis level n_c , yang akan menyebar antara fungsi minimum dan maksimum secara merata.

Euler dapat menandai garis level

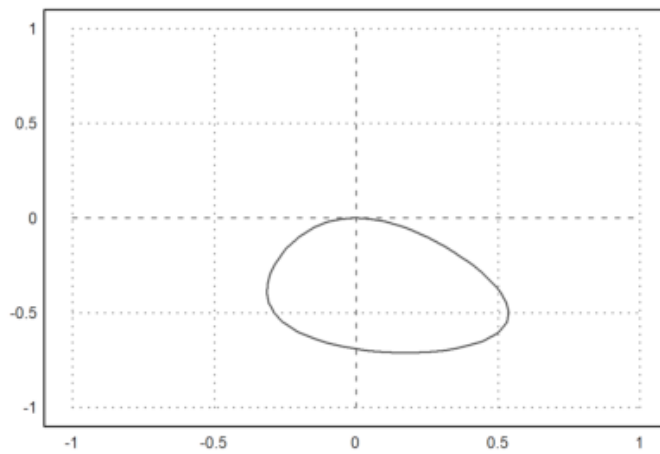
$$f(x,y) = c$$

Untuk menggambar himpunan $f(x,y)=c$ untuk satu atau lebih konstanta c , dapat menggunakan `plot2d()` dengan plot implisitnya di dalam bidang. Parameter untuk c adalah $\text{level}=c$, di mana c dapat berupa vektor garis level.

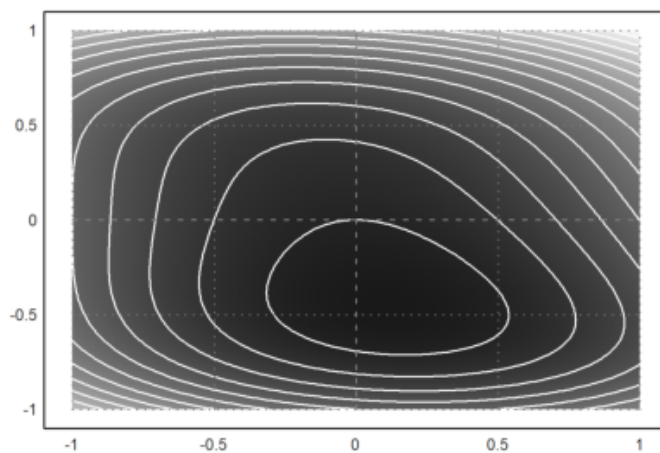
```
>aspect(1.5);
>plot2d("x^2+y^2-x*y-x",r=1.5,level=0, contourcolor=red):
```



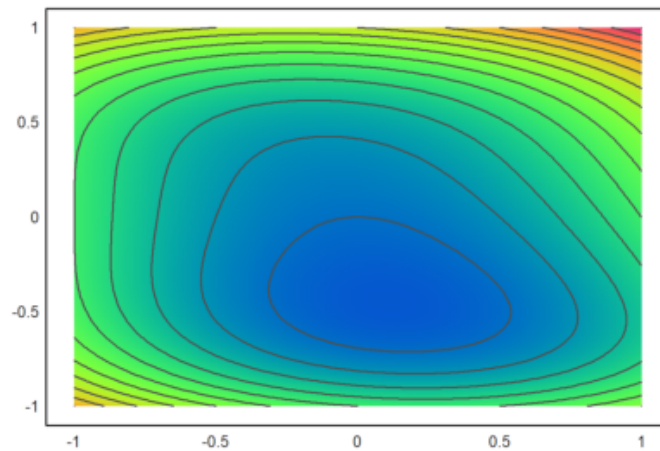
```
>expr := "2*x^2+x*y+3*y^4+y"; // define an expression f(x,y)
>plot2d(expr,level=0): // Solutions of f(x,y)=0;
```



```
>plot2d(expr,level=0:0.5:20,>hue,contourcolor=white,n=200): // nice
```




```
>plot2d(expr,level=0:0.5:20,>hue,>spectral,n=200,grid=4): // nicer
```



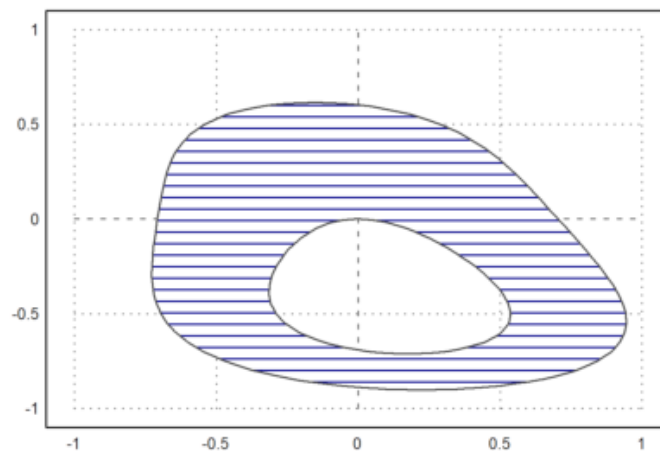
```
>plot2d("x^3-y^2",>contour,>hue,>spectral);
>plot2d("x^3-y^2",level=0,contourwidth=3,>add,contourcolor=red);
>plot2d("x^2+y^3+x*y",level=1,r=4,n=100)
>plot2d("x^2+2*y^2-x*y",level=0:0.1:10,n=100,contourcolor=white,>hue);
```

Juga dimungkinkan untuk mengisi set
lateks: $a \leq f(x,y) \leq b$

dengan rentang tingkat.

Dimungkinkan untuk mengisi wilayah nilai untuk fungsi tertentu. Untuk ini, level harus berupa matriks 2xn. Baris pertama adalah batas bawah dan baris kedua berisi batas atas

```
>plot2d(expr,level=[0;1],style="-",color=blue): // 0 <= f(x,y) <=1;
```



Plot implisit juga dapat menunjukkan rentang level. Kemudian level harus berupa matriks 2xn dari interval level, di mana baris pertama berisi awal dan baris kedua adalah akhir dari setiap interval. Atau, vektor baris sederhana dapat digunakan untuk level, dan parameter dl memperluas nilai level ke interval.

```
>plot2d("x^4+y^4",r=1.5,level=[0;1],color=blue,style="/");
>plot2d("x^2+y^3+x*y",level=[0,2,4;1,3,5],style="/",r=2,n=100);
```

Dimungkinkan juga untuk menandai suatu wilayah

lateks: $a \leq f(x,y) \leq b$.

Ini dilakukan dengan menambahkan level dengan dua baris

```
>plot2d("(x^2+y^2-1)^3-x^2*y^3",r=1.3, ...
>style="##",color=red,<outline, ...
```

#

Space between commands expected!

Found: color=red,<outline, style="##",color=red,<outline, style="##",color=red,<outline, s

You can disable this in the Options menu.

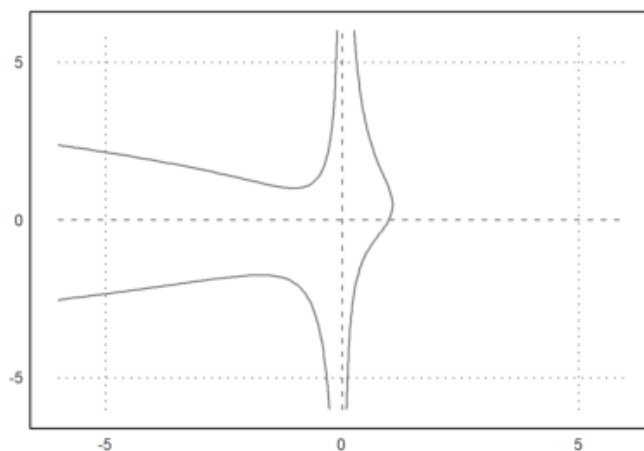
Error in:

```
style="##",color=red,<outline, style="##",color=red,<outline, st ...
^
```

Dimungkinkan untuk menentukan level tertentu. Misalnya, kita dapat memplot solusi persamaan seperti

lateks: $x^3-xy+x^2y^2=6$

```
>plot2d("x^3-x*y+x^2*y^2",r=6,level=1,n=100):
```



Afrilia Sefani(24030130024)

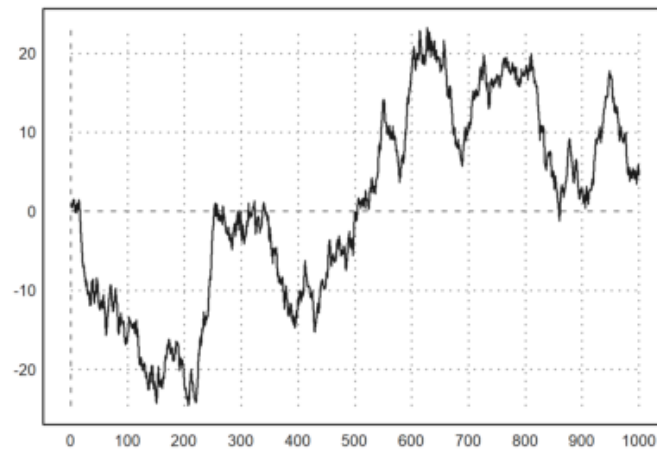
Plot Statistik

Ada banyak fungsi yang dikhususkan pada plot statistik. Salah satu plot yang sering digunakan adalah plot kolom.

A. Jalan Acak (Random Walk)

Jumlah kumulatif dari nilai terdistribusi 0-1-normal menghasilkan jalan acak. Jalan acak adalah model matematika yang menjelaskan jalur yang dibuat oleh serangkaian langkah acak.

```
>plot2d(cumsum(randnormal(1,1000))) :
```



Keterangan:

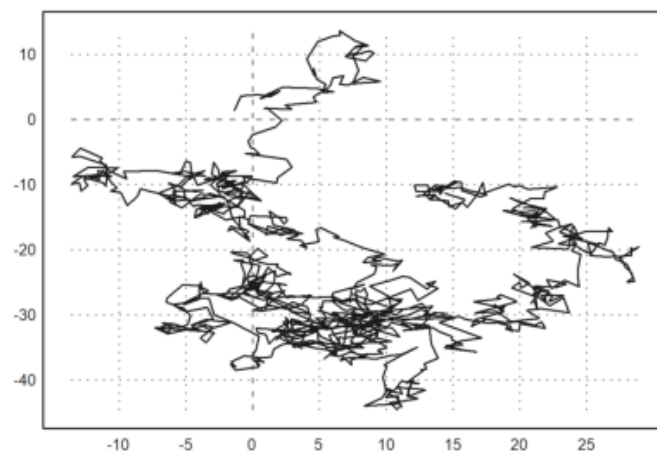
- randn(1,1000): Fungsi ini menghasilkan sebuah baris yang berisi 1000 angka acak yang diambil dari distribusi normal standar.

- cumsum(...): Fungsi ini menghitung jumlah kumulatif angka tersebut.

plot2d(...): Fungsi untuk menggambar hasil jumlah kumulatif tersebut.

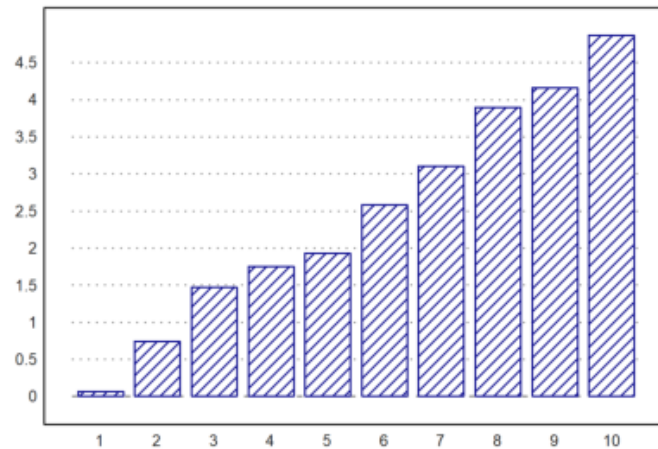
Menggunakan dua baris menunjukkan jalan dalam dua dimensi.

```
>X=cumsum(randnormal(2,1000)); plot2d(X[1],X[2]) :
```



B. Diagram Batang

```
>columnspot(cumsum(random(10)),style="/",color=blue):
```

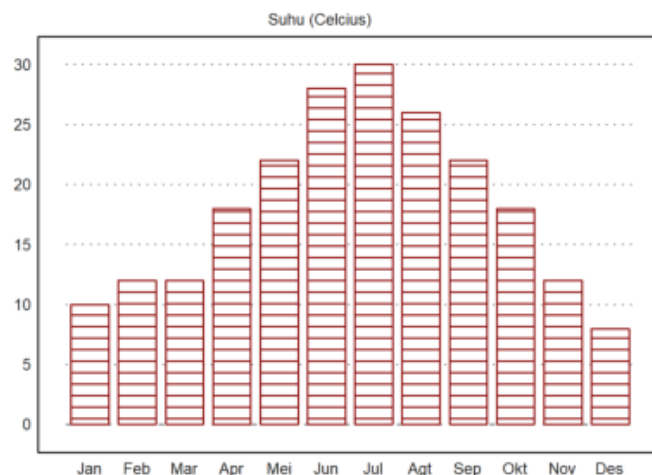


colomnspot(...) adalah perintah untuk menggambar diagram batang.

Hasilnya adalah sebuah diagram batang di mana setiap batang lebih tinggi dari batang sebelumnya karena nilainya adalah jumlah kumulatif.

Selain itu, dapat juga membuat tampilan yang lebih informatif dengan menambahkan label.

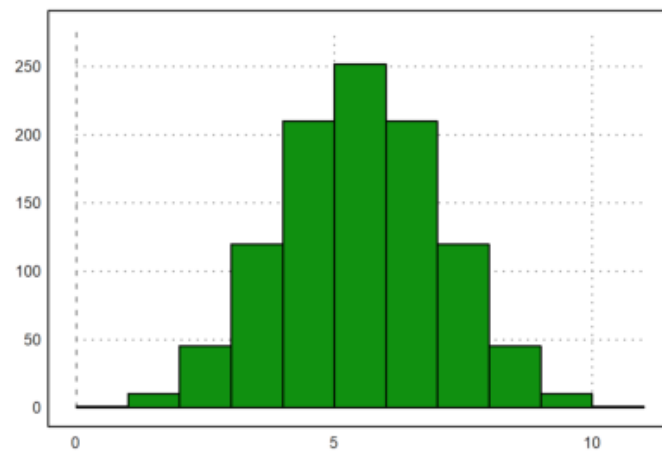
```
>months=["Jan","Feb","Mar","Apr","Mei","Jun","Jul","Agt","Sep","Okt","Nov","Des"];  
>values=[10,12,12,18,22,28,30,26,22,18,12,8];  
>columnspot(values,lab=months,color=red,style="-");  
>title("Suhu (Celcius)");
```



Keterangan:

- values: menyediakan data untuk tinggi setiap batang
- lab=months: menggunakan 'months' sebagai label untuk setiap batang pada sumbu x.
- title("Temperature"): perintah menambahkan judul "Temperature" di atas grafik.

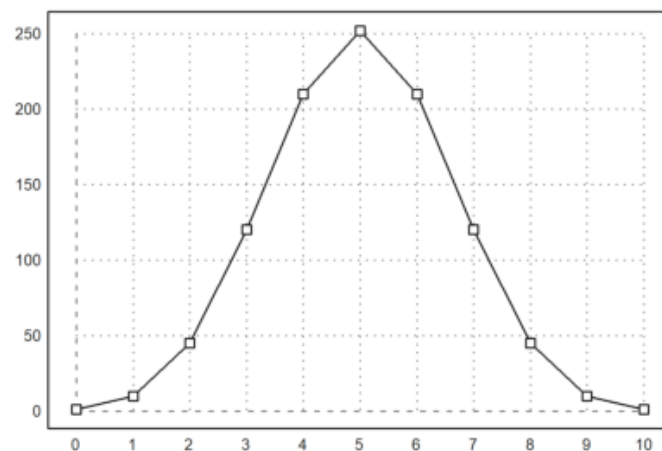
```
>k=0:10;
>plot2d(k,bin(10,k),>bar):
```



Keterangan:

- bin(10,k) adalah fungsi yang menghitung koefisien binomial (sering ditulis $C(n,k)$).
- >bar memerintahkan program untuk membuat grafik batang.

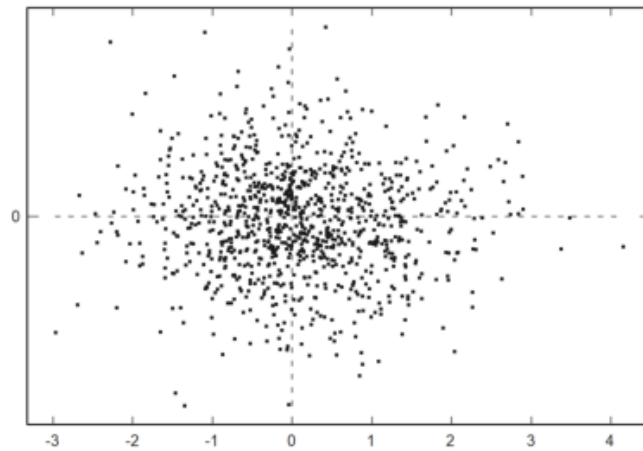
```
>plot2d(k,bin(10,k));
>plot2d(k,bin(10,k),>points,>add):
```



Keterangan:

- >points untuk memerintahkan program menampilkan titik-titik data.
- >add untuk memastikan bahwa plot baru ditambahkan ke grafik yang sudah ada.

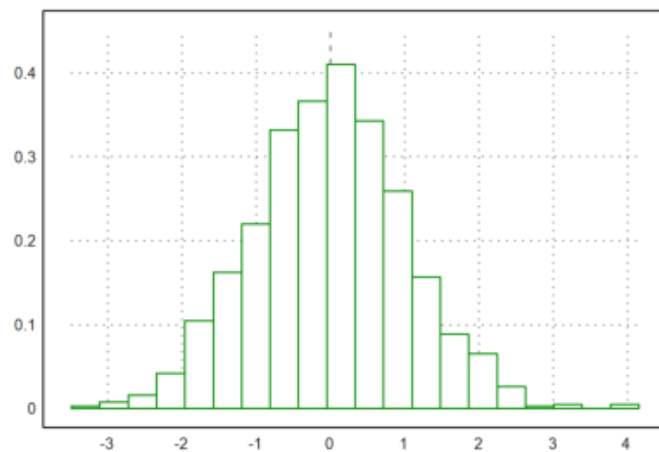
```
>plot2d(normal(1000),normal(1000),>points,grid=6,style=".."):
```



Keterangan:

- >points untuk memerintahkan program menggambar titik-titik saja.
- grid=6 untuk menentukan jumlah petak pada grid yang ditampilkan.

```
>plot2d(normal(1,1000),>distribution,style="O") :
```

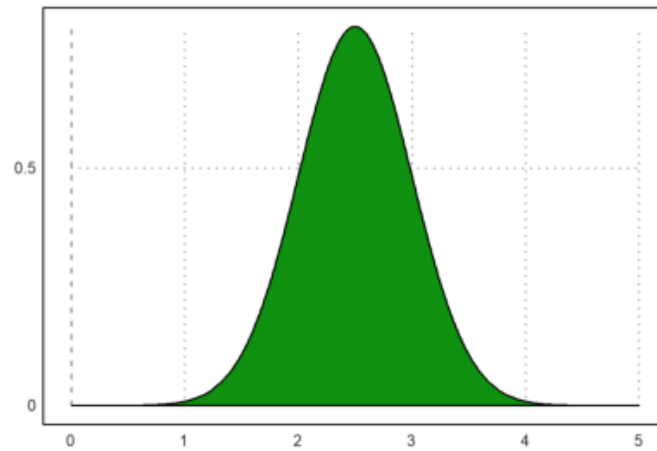


Keterangan:

- >distribution memberi tahu program untuk membuat histogram dari data yang dihasilkan.
- style"O" membuat batang-batang terlihat seperti kotak-kotak kosong

C. Distribusi Normal

```
>plot2d("qnormal",0,5;2.5,0.5,>filled) :
```

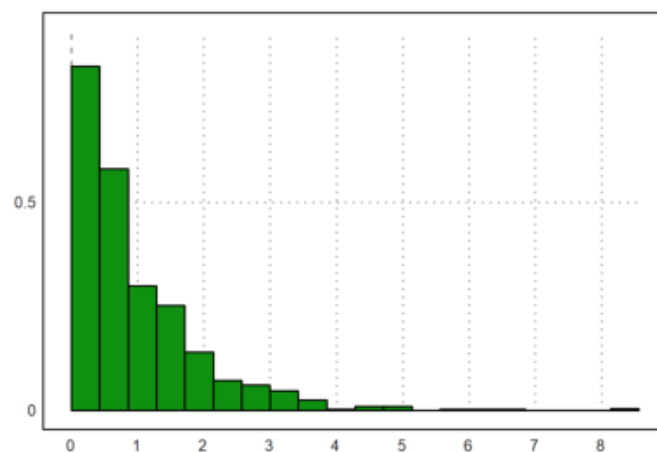


Keterangan:

- gnorm adalah nama fungsi yang akan di plot. Di sini gnorm adalah kepadatan probabilitas.
- 0,5 adalah rentang sumbu x
- 2.5 adalah mean dari distribusi yang menentukan di mana puncak berada. Maka puncaknya berada di $x=2.5$
- 0,5 untuk standar deviasi atau seberapa lebar kurva.

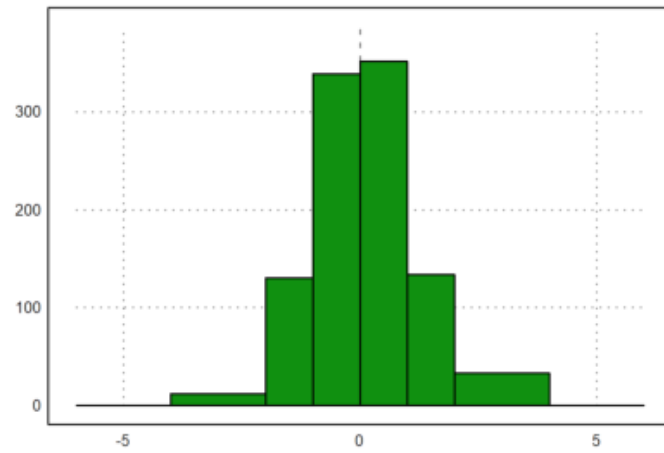
Untuk memplot distribusi statistik eksperimental, dapat menggunakan `distribution=n` dengan `plot2d`.

```
>w=randexponential(1,1000); // exponential distribution
>plot2d(w,>distribution): // or distribution=n with n intervals
```



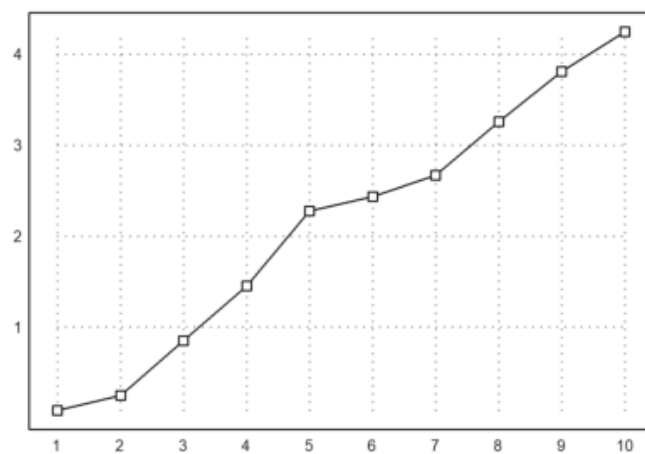
Atau Anda dapat menghitung distribusi dari data dan memplot hasilnya dengan `>bar` di `plot3d`, atau dengan `plot kolom`.

```
>w=normal(1000); // 0-1-normal distribution
>{x,y}=histo(w,10,v=[-6,-4,-2,-1,0,1,2,4,6]); // interval bounds
>plot2d(x,y,>bar):
```

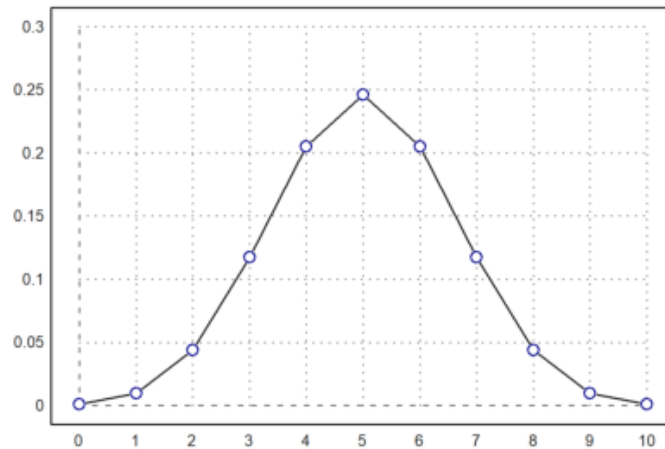


Fungsi `statplot()` mengatur gaya dengan string sederhana.

```
>statplot(1:10,cumsum(random(10)),"b"):
```



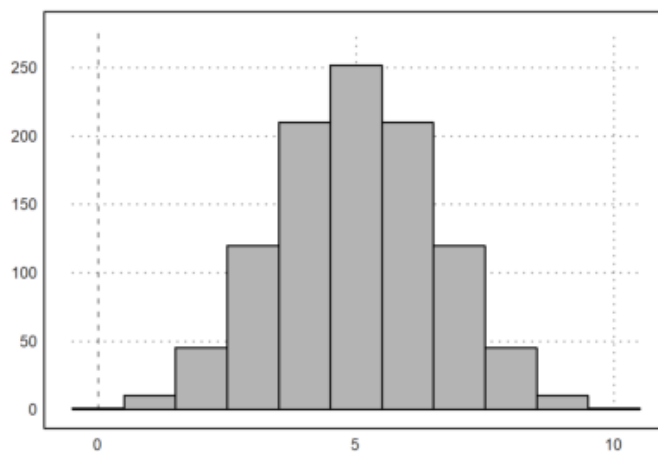
```
>n=10; i=0:n; ...
>plot2d(i,bin(n,i)/2^n,a=0,b=10,c=0,d=0.3); ...
>plot2d(i,bin(n,i)/2^n,points=true,style="ow",add=true,color=blue):
```

Selain itu, data dapat diplot sebagai batang. Dalam hal ini, x harus diurutkan dan satu elemen lebih panjang dari y. Batang akan memanjang dari $x[i]$ hingga $x[i+1]$ dengan nilai $y[i]$. Jika x berukuran sama dengan y, x akan memanjang satu elemen dengan spasi terakhir.

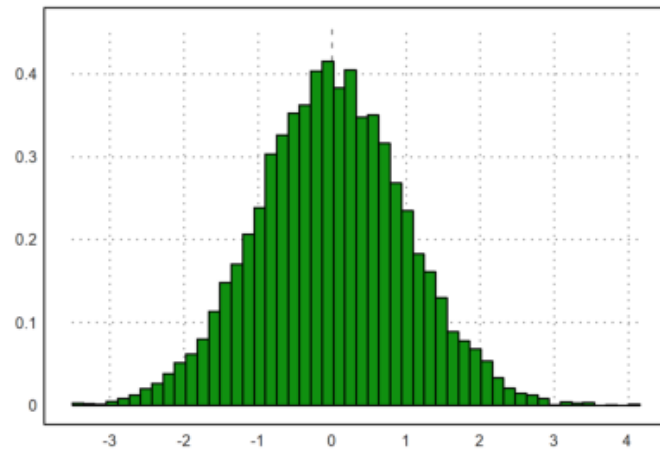
Gaya isian dapat digunakan seperti di atas.

```
>n=10; k=bin(n,0:n); ...
>plot2d(-0.5:n+0.5,k,bar=true,fillcolor=lightgray):
```

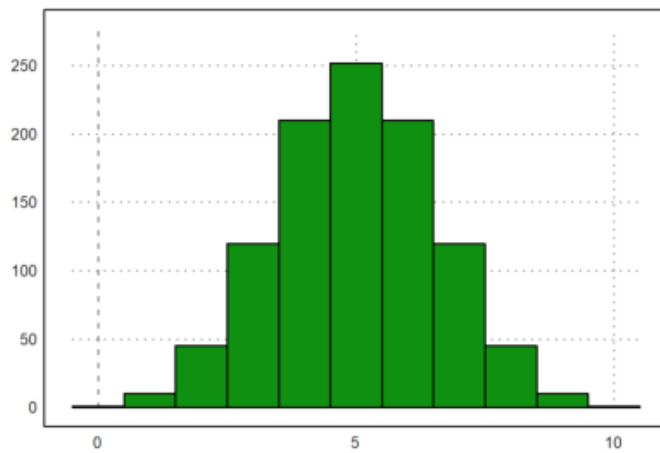


Data untuk diagram batang (batang=1) dan histogram (histogram=1) dapat diberikan secara eksplisit dalam xv dan yv, atau dapat dihitung dari distribusi empiris dalam xv dengan distribusi > (atau distribusi=n). Histogram nilai xv akan dihitung secara otomatis dengan histogram >. Jika >genap ditentukan, nilai xv akan dihitung dalam interval bilangan bulat.

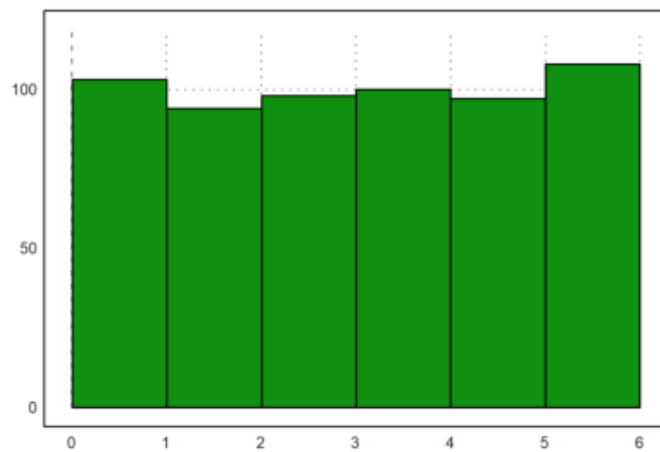
```
>plot2d(normal(10000),distribution=50):
```



```
>k=0:10; m=bin(10,k); x=(0:11)-0.5; plot2d(x,m,>bar):
```

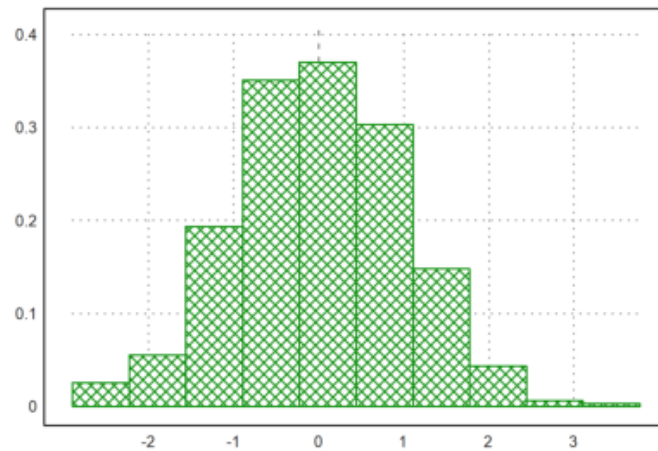


```
>plot2d(random(600)*6,histogram=6):
```



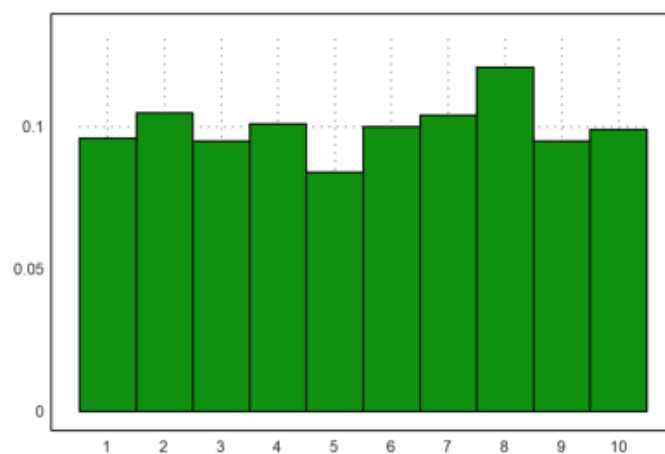
Untuk distribusi, ada parameter `distribution=n`, yang menghitung nilai secara otomatis dan mencetak distribusi relatif dengan `n` sub-interval.

```
>plot2d(normal(1,1000),distribution=10,style="\/"):
```



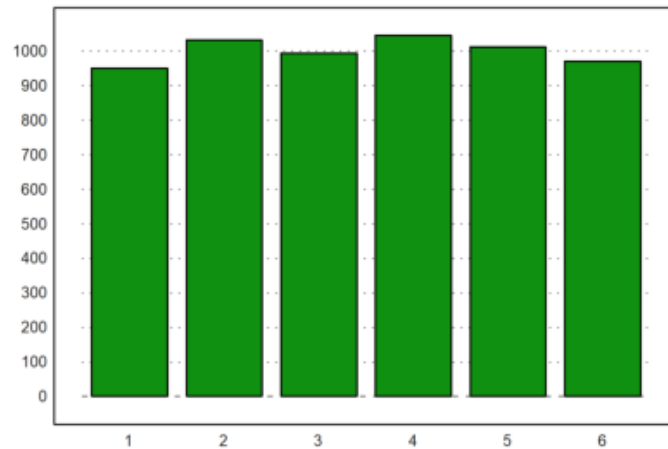
Dengan parameter `even=true`, ini akan menggunakan interval integer.

```
>plot2d(intrandom(1,1000,10),distribution=10,even=true):
```

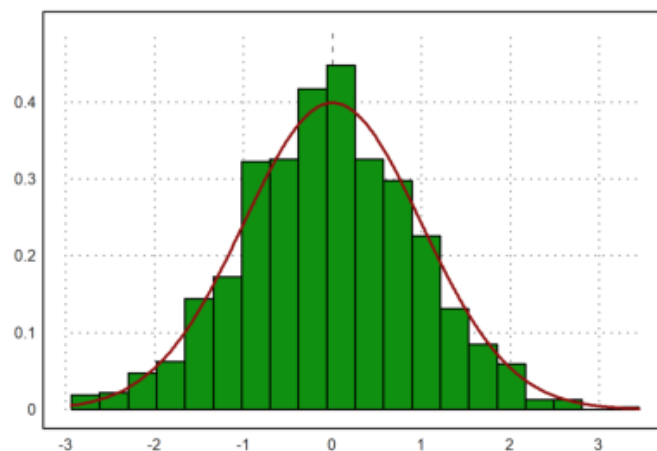


Perlu diketahui bahwa ada banyak plot statistik yang mungkin berguna. Lihat tutorial tentang statistik.

```
>columnplot(getmultiplicities(1:6,intrandom(1,6000,6))):
```



```
>plot2d(normal(1,1000),>distribution); ...
>plot2d("qnormal(x)",color=red,thickness=2,>add):
```



Keterangan:

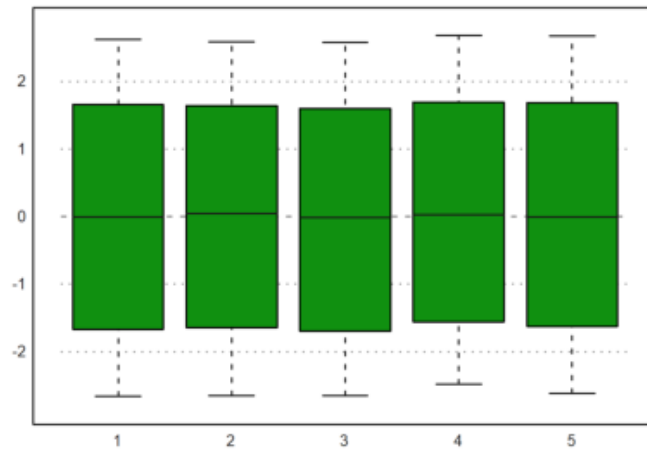
- normal(1,1000): menampilkan 1000 angka acak dari distribusi normal standar.
- >distribution: membuat histogram dari data tersebut. Histogram tersebut merepresentasikan distribusi empiris.
- gnormal(x): untuk menggambar kurva lonceng (kurva distribusi normal). Kurva tersebut merepresentasikan distribusi empiris.
- thickness=2 untuk membuat kurva menjadi lebih tebal.
- >add: untuk memerintahkan program untuk menambahkan kurva ke grafik yang sudah ada.

Grafik yang dihasilkan menunjukkan bahwa distribusi empiris dari data acak sangat mirip dengan distribusi teoretis yang ideal.

Ada juga banyak plot khusus untuk statistik. Boxplot menunjukkan kuartil dari distribusi ini dan banyak outlier. Menurut definisi, outlier dalam boxplot adalah data yang melebihi 1,5 kali kisaran 50% tengah plot.

D. Boxplot

```
>M=normal(5,1000); boxplot(quartiles(M)):
```



Keterangan:

- `M=normal(5,1000)`: membuat matriks M berisi angka acak dari distribusi normal dengan 5 menunjukkan 5 kolom dan 1000 menunjukkan 1000 baris
- `quartiles(M)`: menghitung kuartil

Afrilia Sefani (24030130024)

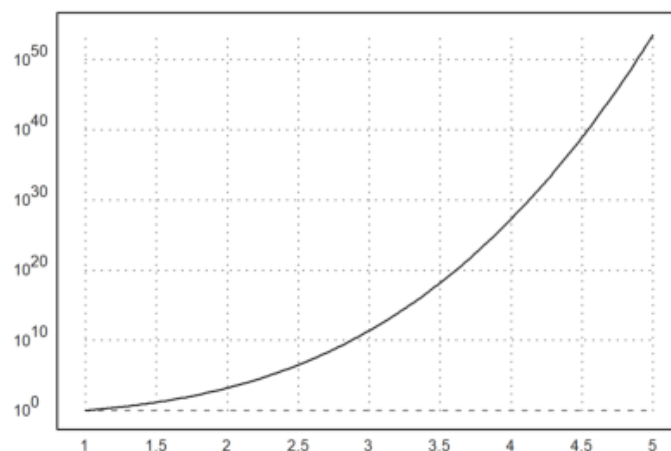
Plot Logaritmik

EMT menggunakan parameter "logplot" untuk skala logaritmik.

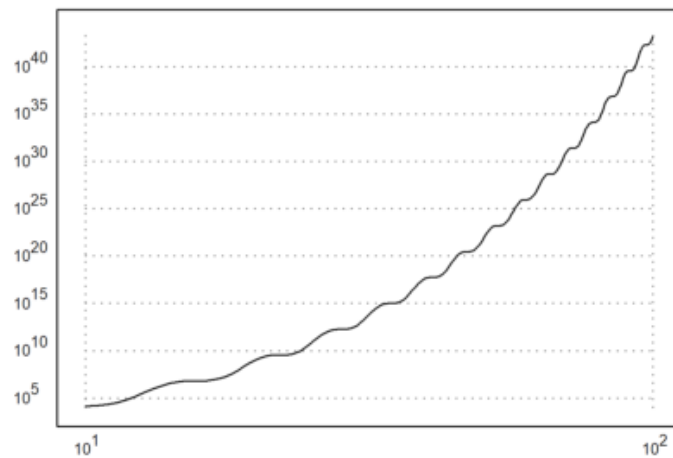
Plot logaritma dapat diplot baik menggunakan skala logaritma dalam y dengan `logplot=1`, atau menggunakan skala logaritma dalam x dan y dengan `logplot=2`, atau dalam x dengan `logplot=3`.

- `logplot=1`: y-logaritma, artinya mengubah sumbu y menjadi skala logaritmik, sementara sumbu x tetap linear.
- `logplot=2`: x-y-logaritma, artinya mengubah kedua sumbu (x dan y) menjadi skala logaritmik.
- `logplot=3`: x-logaritma, artinya mengubah sumbu x menjadi skala logaritmik.

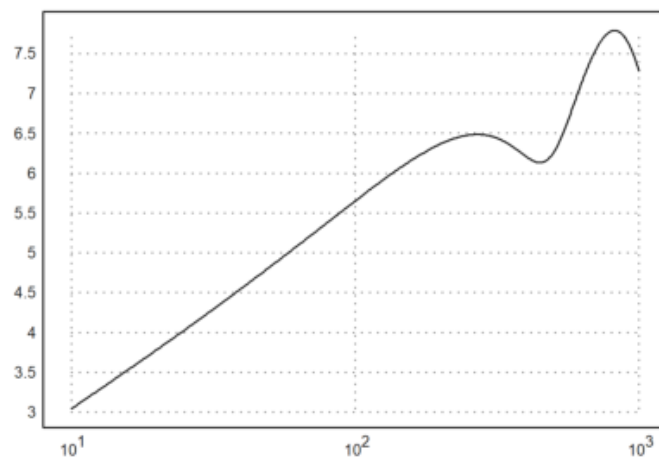
```
>plot2d("exp(x^3-x)*x^2",1,5,logplot=1):
```



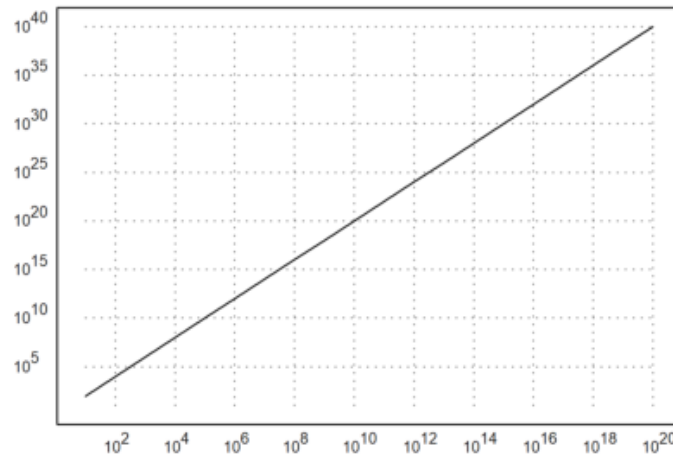
```
>plot2d("exp(x+sin(x))",10,100,logplot=2):
```



```
>plot2d("log(x*(2+sin(x/100)))",10,1000,logplot=3):
```



```
>x=10^(1:20); y=x^2-x;  
>plot2d(x,y,logplot=2):
```



Afrilia Sefani (24030130024)

Menggambar Segibanyak

Fungsi perintah yang digunakan untuk menggambar segi banyak atau poligon, yaitu.

```
x=linspace(2,2pi,n);plot2d(cos(x),sin(x),r=1,>filled,style="..."):
```

atau dengan fungsi perintah berikut

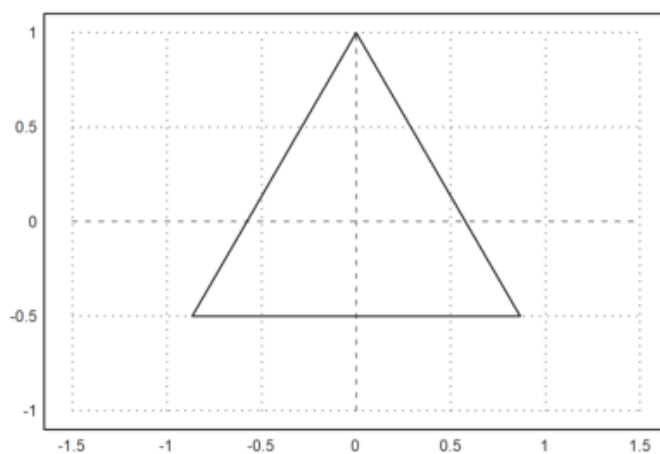
```
x=linspace(0,2pi,n);plot2d(sin(x),cos(x),r=1,>filled,styles="...",fillcolor=red):
```

Keterangan:

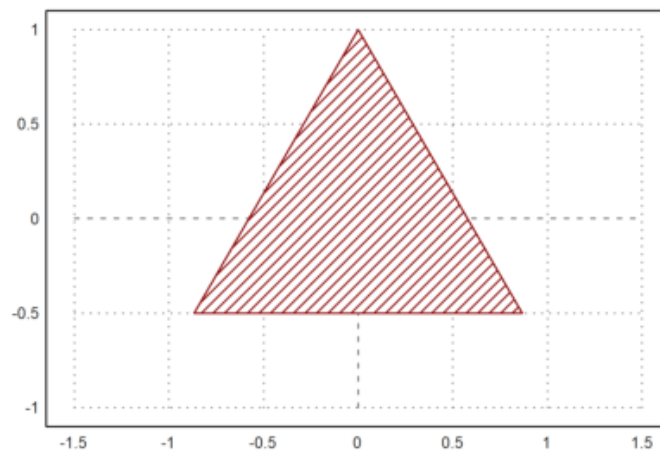
- filled=true untuk mengisi plot
- styles"..." untuk memilih gaya plot, dapat "/", "\" atau lainnya.
- fillcolor: untuk memberikan warna

A. Menggambar Segitiga

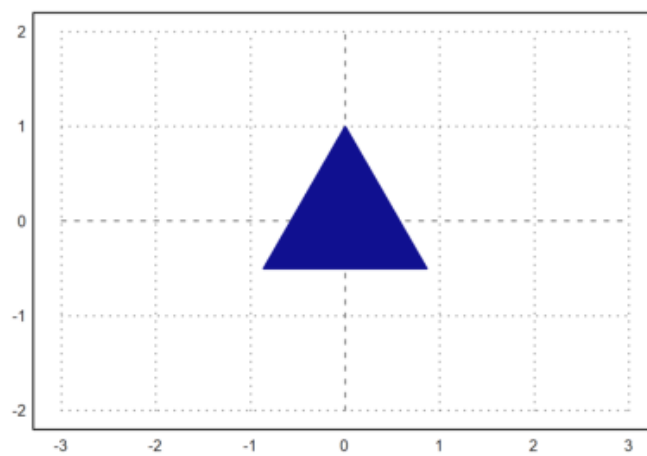
```
>x=linspace(0,2pi,3);...
>plot2d(sin(x),cos(x),r=1):
```



```
>x=linspace(0,2pi,3);...
>plot2d(sin(x),cos(x),>filled,style="/",fillcolor=red,r=1):
```

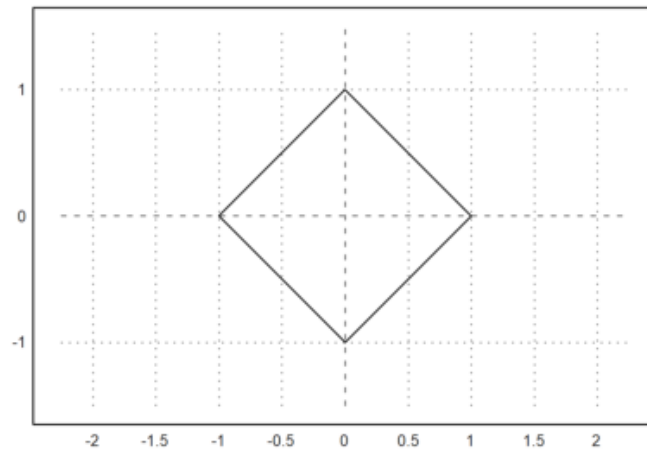


```
>x=linspace(0,2pi,3);...
>plot2d(sin(x),cos(x),>filled,style="#",fillcolor=blue,r=2):
```

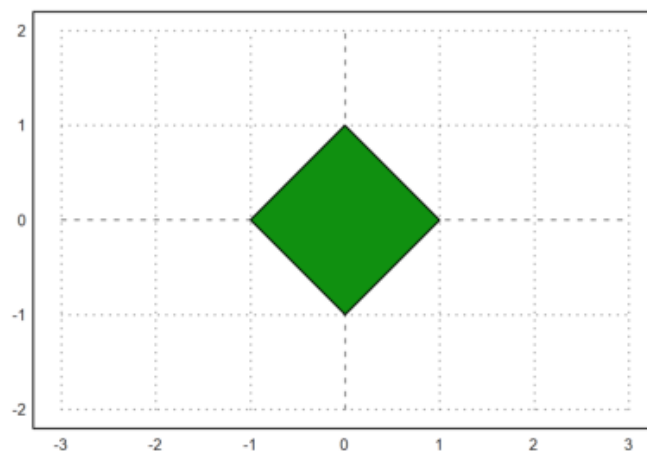


B. Menggambar Segiempat

```
>x=linspace(0,2pi,4);...
>plot2d(cos(x),sin(x),r=1.5):
```

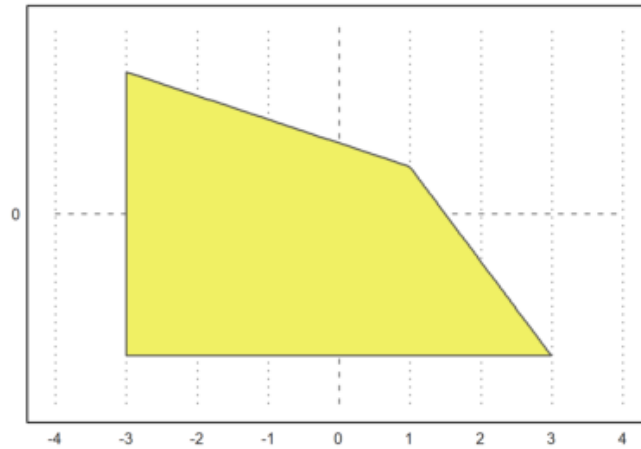



```
>x=linspace(0,2pi,4);
>plot2d(cos(x),sin(x),r=2,>filled,outline=1):
```



Selain menggubakan fungsi tersebut, dapat juga menggunakan fungsi maksimum. Fungsi maksimum digunakan untuk menggabungkan semua batasan linear (garis-garis batas) menjadi satu fungsi tunggal yang dapat diplot, sehingga menciptakan bentuk poligon.

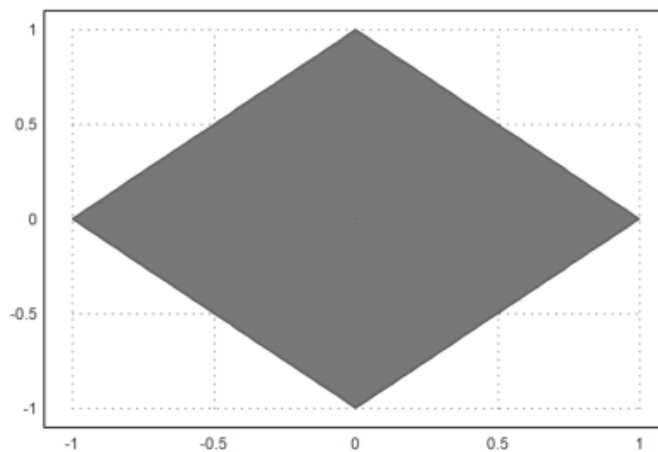
```
>A=[2,1;1,2;-1,0;0,-1];
>function f(x,y):=max([x,y].A');
>plot2d("f",r=4,level=[0;3],color=yellow,n=111):
```



Keterangan:

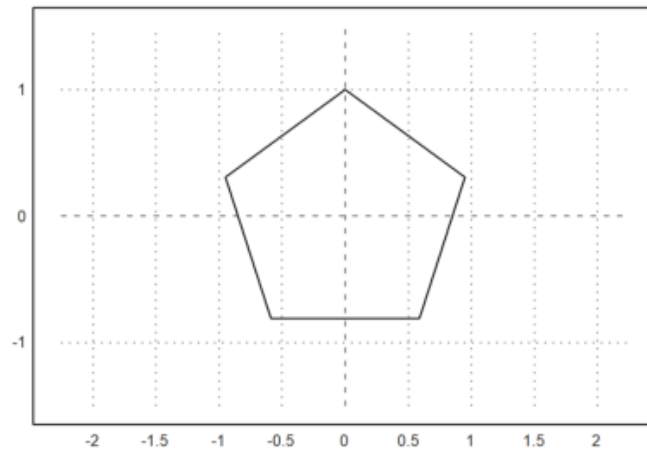
- A adalah matriks A. Setiap garis dalam matriks ini adalah vektor normal dari garis-garis yang membentuk sisi poligon. Vektor ini mewakili koefisien x dan y dari setiap pertidaksamaan.
- Fungsi $f(x,y)$ menghitung nilai maksimum dari serangkaian ekspresi linear.
- $[x,y].A'$ melakukan perkalian matriks yang setara dengan menghitung nilai $ax+by$ untuk setiap baris di matriks A.
- $r=4$ menetapkan rentang koordinat x dan y
- $level=[0;3]$ merupakan parameter yang menentukan rentang nilai fungsi yang akan diwarnai.
- n adalah parameter yang menentukan jumlah titik data yang digunakan untuk menggambar plot. Jika n kecil, jarak antar titik yang dihitung menjadi terlalu jauh, akibatnya sudut-sudut poligon yang seharusnya tajam menjadi terlihat tumpul.

```
>A=[1,1;-1,1;-1,-1;1,-1];
>function f(x,y) :=max([x,y].A');
>plot2d("f",r=1,level=[0;1],color=gray,n=90):
```



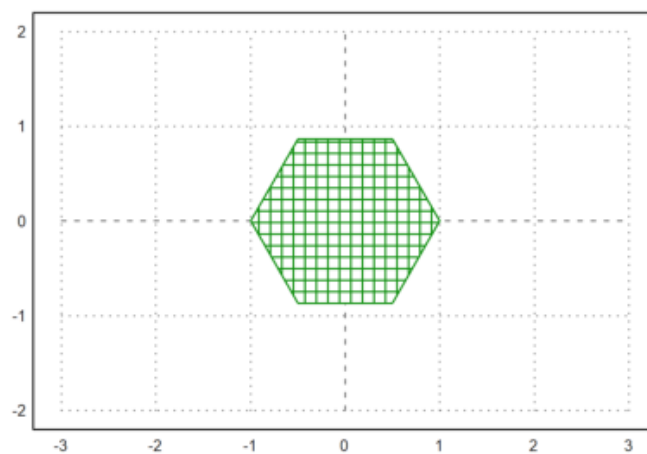
C. Menggambar Segilima

```
>t=linspace(0,2pi,5); plot2d(sin(t),cos(t),r=1.5):
```



D. Menggambar Heksagon

```
>t=linspace(0,2pi,6);...
>plot2d(cos(t),sin(t),>filled,style="+",fillcolor=green,r=2):
```



E. Menggambar Dekagon

```
>t=linspace(0,2pi,10);...
>plot2d(cos(t),sin(t),>filled,style="-", fillcolor=red,r=2):
```

