EMT untuk Perhitungan Aljabar

Afrilia Sefani (24030130024)

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))))
```

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

Baris Perintah

Sebuah baris perintah pada Euler terdiri dari satu atau beberapa perintah Euler yang diikuti oleh tanda titik koma ";" atau koma ",". Titik koma digunakan untuk mencegah hasil ditampilkan. Koma setelah perintah terakhir boleh dihilangkan.

Baris perintah berikut hanya akan menampilkan hasil dari ekspresi, bukan hasil dari penugasan atau perintah format.

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

```
    16.76
```

Perintah-perintah harus dipisahkan dengan spasi. Baris perintah berikut akan menampilkan dua hasilnya.

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

```
     50.27
    100.53
```

Baris perintah dijalankan sesuai urutan ketika pengguna menekan Enter/Return. Jadi, kamu akan mendapatkan nilai baru setiap kali menjalankan baris kedua.

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

```
    0.54
```

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

```
    0.86
```

Jika dua baris dihubungkan dengan "...", maka keduanya akan selalu dijalankan secara bersamaan.

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

```
    1.42
    1.41
    1.41
```

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

Untuk melipat semua multi-line, tekan Ctrl+L. Maka baris-baris berikutnya hanya akan terlihat jika salah satunya mendapat fokus. Untuk melipat satu multi-line saja, mulai baris pertama dengan "%+".

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

Sebuah baris yang diawali dengan %% akan benar-benar tidak terlihat.

```
    81.00
```

Euler mendukung penggunaan perulangan (loops) di dalam baris perintah, selama masih muat dalam satu baris atau milti-line. Dalam program, tentu saja batasan ini 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
```

```
1.50
1.42
1.41
1.41
1.41
```

Penggunaan multi-line diperbolehkan. Pastikan baris diakhiri dengan "...".

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

```
1.41
```

Struktur percabangan (kondisional) juga dapat digunakan.

```
>if E^pi>pi^E; then "Thought so!", endif;
```

```
Thought so!
```

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

Ketika kursor digerakkan sepanjang baris, pasangan tanda kurung buka dan kurung tutup akan disorot. Perhatikan juga status line. Setelah tanda kurung buka dari fungsi sqrt(), status line akan menampilkan teks bantuan untuk fungsi tersebut. Jalankan perintah dengan tombol Enter/Return.

```
>sqrt(sin(10°)/cos(20°))
```

```
0.43
```

Untuk melihat bantuan dari perintah terakhir, buka jendela bantuan dengan tombol F1. Di sana, Anda bisa mengetik teks untuk dicari. Pada baris kosong, akan muncul bantuan tentang jendela bantuan itu sendiri. Anda dapat menekan Escape untuk menghapus baris atau menutup jendela bantuan.

Anda juga dapat double click pada perintah apa pun untuk membuka bantuan terkait perintah tersebut. Coba lakukan double click pada perintah exp di bawah ini pada baris perintah.

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

```
2.50
```

Anda juga dapat melakukan copy-paste di Euler. Gunakan Ctrl+C dan Ctrl+V. Untuk menandai teks, tarik dengan mouse atau gunakan Shift bersamaan dengan tombol panah. Selain itu, Anda juga bisa menyalin tanda kurung yang sedang disorot.

Euler mengenal fungsi-fungsi matematika umum. Seperti yang sudah Anda lihat sebelumnya, fungsi trigonometri dapat bekerja dalam radian atau derajat. Untuk mengubah ke derajat, tambahkan simbol derajat (dengan tombol F7) pada nilai, atau gunakan fungsi rad(x). Fungsi akar kuadrat ditulis dengan sqrt di Euler. Tentu saja, bentuk x^(1/2) juga memungkinkan.

Untuk mendefinisikan variabel, gunakan tanda "=" atau ":=". Demi kejelasan, pengantar ini menggunakan bentuk kedua (:=). Spasi sebenarnya tidak masalah, tetapi spasi antar perintah diharapkan ada.

Beberapa perintah dalam satu baris dipisahkan dengan "," atau ";". Tanda titik koma (;) digunakan untuk menyembunyikan hasil perintah. Jika pada akhir baris tidak ada titik koma, maka akan dianggap ada koma.

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

```
30.66
```

EMT menggunakan sintaks pemrograman untuk ekspresi. Untuk menuliskan

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

Anda harus menuliskan tanda kurung dengan benar dan menggunakan / untuk pecahan. Perhatikan tanda kurung yang disorot sebagai bantuan. Ingat bahwa konstanta Euler e ditulis sebagai E di EMT.

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

```
8.78
```

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 menuliskannya dalam bentuk baris lurus (line form)

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

```
23.27
```

Tempatkan tanda kurung dengan hati-hati pada sub-ekspresi yang harus dihitung terlebih dahulu. EMT membantu Anda dengan menyoroti bagian ekspresi yang ditutup oleh tanda kurung. Anda juga harus mengetikkan kata "pi" untuk huruf Yunani p.

Hasil dari perhitungan ini berupa bilangan pecahan desimal (floating point number). Secara bawaan, hasil ditampilkan dengan ketelitian sekitar 12 digit. Pada contoh perintah berikut, kita juga belajar bagaimana merujuk hasil sebelumnya dalam baris yang sama.

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

```
0.48
10/21
```

Sebuah perintah di Euler dapat berupa ekspresi atau perintah dasar (primitive command). Ekspresi tersusun dari operator dan fungsi. Jika perlu, ekspresi harus berisi tanda kurung untuk memaksa urutan perhitungan yang benar. Jika ragu, menambahkan tanda kurung adalah ide yang baik. Perlu dicatat bahwa EMT menyorot pasangan kurung buka dan kurung tutup saat mengedit baris perintah.

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

```
        14.50
```

Operator numerik di Euler meliputi:

+ plus (unary atau operator)
- minus (unary atau operator)
*, / untuk perkalian dan pembagian
. untuk perkalian matriks
a^b untuk perpangkatan jika a positif atau b bilangan bulat (bentuk a**b juga bisa digunakan)
n! operator faktorial

dan masih banyak lagi.

Berikut beberapa fungsi yang mungkin Anda perlukan. Masih ada banyak lagi selain ini.

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

Beberapa perintah memiliki alias, misalnya ln untuk log.

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

```
         2.00
         0.50
```

```
>sin(30°)
```

```
         0.50
```

Pastikan menggunakan tanda kurung (kurung biasa) setiap kali ada keraguan tentang urutan operasi! Contoh berikut tidak sama dengan (2^3)^4, yang merupakan bentuk default untuk 2^3^4 di EMT (beberapa sistem numerik lain melakukannya dengan cara berbeda.

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

```
  2417851639229258349412352.00
      4096.00
  2417851639229258349412352.00
```

Bilangan Riil

Tipe data utama di Euler adalah bilangan riil. Bilangan riil direpresentasikan dalam format IEEE dengan ketelitian sekitar 16 digit desimal.

```
>longest 1/3
```

```
      0.3333333333333333
```

Representasi internal ganda memerlukan 8 byte.

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

```
  1.0101010101010101010101010101010101010101010101010101*2^-2
```

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

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

Teks

Sebuah string di Euler didefinisikan dengan tanda "...".

```
>"A string can contain anything."
```

```
  A string can contain anything.
```

String dapat berisi apa saja.
tring dapat digabungkan dengan | atau dengan +. Ini juga bisa bekerja dengan angka, yang dalam hal ini akan dikonversi menjadi string.

```
>"The area of the circle with radius " + 2 + " cm is " + pi*4 + " cm^2."
```

```
  The area of the circle with radius 2 cm is 12.5663706144 cm^2.
```

Fungsi print juga bisa mengubah angka menjadi string. Fungsi ini dapat menerima parameter jumlah digit, jumlah tempat (0 untuk keluaran rapat), dan opsional satuan.

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

```
  Golden Ratio : 1.61803
```

Ada sebuah string khusus yaitu none, yang tidak akan ditampilkan. String ini dikembalikan oleh beberapa fungsi ketika hasilnya tidak penting. (String ini juga dikembalikan secara otomatis jika suatu fungsi tidak memiliki pernyataan return.)

```
>none
```

Untuk mengubah sebuah string menjadi angka, cukup evaluasi string tersebut. Ini juga berlaku untuk ekspresi (lihat di bawah).

```
>"1234.5"()
```

```
      1234.50
```

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

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

```
  affe
  charlie
  bravo
```

Vektor string kosong ditandai dengan [none]. Vektor string dapat digabungkan.

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

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

String dapat berisi karakter Unicode. Secara internal, string ini berisi kode UTF-8. Untuk membuat string seperti itu, gunakan u"..." bersama dengan salah satu HTML entity.
String Unicode dapat digabungkan seperti string lainnya.

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

```
   = 45°
```

I

Di dalam komentar, entity yang sama seperti , °, dan sebagainya juga dapat digunakan. Ini bisa menjadi alternatif cepat untuk LaTeX. (Detail lebih lanjut tentang komentar ada di bagian berikutnya.

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

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

```
  Real 1 x 20 matrix

       196.00        32.00       105.00       115.00     ...
```

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

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

```
Ü is a German letter
```

Fungsi utf() dapat menerjemahkan string dengan entity di dalam sebuah variabel menjadi string Unicode.

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

```
We have =.
```

Selain itu, dimungkinkan juga menggunakan entity numerik.

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

```
Ähnliches
```

Nilai Boolean

Nilai boolean direpresentasikan dengan 1 = true (benar) atau 0 = false (salah) di Euler. String dapat dibandingkan sama seperti angka.

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

```
0.00
1.00
```

Operator "and" ditulis dengan && dan "or" ditulis dengan ||, sama seperti pada bahasa C. (Kata "and" dan "or" hanya bisa digunakan dalam kondisi pada perintah if).

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

```
1.00
```

Operator boolean mengikuti aturan bahasa matriks di Euler.

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

```
Real 1 x 10 matrix

0.00        0.00        0.00        0.00     ...
6.00        7.00        8.00        9.00       10.00
```

Anda dapat menggunakan fungsi nonzeros() untuk mengekstrak elemen-elemen tertentu dari sebuah vektor. Pada contoh berikut, digunakan syarat isprime(n) (bilangan prima).

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

```
Real 1 x 50 matrix

       2.00         3.00         5.00         7.00     ...
```

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

```
Real 1 x 25 matrix

       2.00         3.00         5.00         7.00     ...
```

Format Keluaran

Format keluaran bawaan dari EMT menampilkan 12 digit. Untuk memastikan kita melihat format bawaan, kita reset formatnya.

```
>defformat; pi
```

```
3.14159265359
```

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

```
>longest pi
```

```
      3.141592653589793
```

Berikut adalah representasi internal dalam bentuk heksadesimal dari sebuah bilangan double.

```
>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
```

Format bawaan adalah format(12).

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

```
0.333333333333
```

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

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

```
    0.52    0.51    0.84    0.61    0.37    0.36    0.93    0.85
    0.76    0.17    0.87    0.98   0.091    0.96    0.39    0.34
    0.77    0.86    0.84    0.66    0.74    0.95    0.35   0.048
```

Format bawaan untuk skalar adalah format(12). Namun ini bisa diubah.

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

```
3.1416
```

Fungsi "longestformat" juga mengatur format skalar.

```
>longestformat; pi
```

```
3.141592653589793
```

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

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

Ketelitian internal EMT adalah sekitar 16 digit desimal, sesuai standar IEEE. Angka disimpan dalam format internal ini.

```
>longestformat; pi,
```

```
3.141592653589793
```

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

```
  3.14159
```

Format bawaan adalah defformat().

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

Ada operator singkat yang hanya mencetak satu nilai. Operator "longest" akan menampilkan semua digit valid dari sebuah bilangan.

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

```
        4.934802200544679
```

Ada juga operator singkat untuk menampilkan hasil dalam bentuk pecahan. Kita sudah menggunakannya sebelumnya.

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

```
  25/12
```

Karena format internal menggunakan cara biner untuk menyimpan angka, nilai 0.1 tidak bisa direpresentasikan dengan tepat. Kesalahannya akan bertambah sedikit demi sedikit, seperti terlihat pada 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
```

Namun dengan format bawaan "longformat" Anda tidak akan menyadari hal ini. Untuk kenyamanan, hasil keluaran dari bilangan yang sangat kecil dianggap 0.

```
>0.1+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 bisa dievaluasi oleh EMT. Untuk itu, gunakan tanda kurung setelah ekspresi. Jika Anda bermaksud menggunakan string sebagai ekspresi, gunakan konvensi penamaan seperti "fx" atau "fxy", dan seterusnya. Ekspresi memiliki prioritas lebih tinggi dibandingkan fungsi.

Variabel global dapat digunakan dalam evaluasi ekspresi.

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

```
        12.56637061435917
```

Parameter secara otomatis diberikan pada x, y, dan z sesuai urutan tersebut. Parameter tambahan dapat ditambahkan dengan cara menyisipkan parameter yang ditetapkan.

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

```
  -0.919535764538
```

Perlu dicatat bahwa ekspresi selalu menggunakan variabel global, meskipun ada variabel di dalam sebuah fungsi dengan nama yang sama. (Jika tidak demikian, evaluasi ekspresi di dalam fungsi bisa menghasilkan hasil yang sangat membingungkan bagi pengguna yang memanggil fungsi tersebut.)

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

```
36
```

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

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

```
45
```

Sebagai referensi, perlu dicatat bahwa kumpulan pemanggilan (dibahas di bagian lain) dapat berisi ekspresi. Jadi, contoh di atas dapat dibuat 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.Perlu dicatat bahwa mendefinisikan sebuah fungsi dengan nama yang sama seperti ekspresi simbolik global akan menghapus variabel ini untuk menghindari kebingungan antara ekspresi simbolik dan fungsi.

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

```
12
```

Sebagai konvensi, ekspresi simbolik atau numerik sebaiknya dinamai fx, fxy, dll., dan skema penamaan ini tidak boleh digunakan untuk fungsi.

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

$$x^x\,\left(\log x+1\right)$$

Bentuk khusus dari sebuah ekspresi memungkinkan setiap variabel digunakan sebagai parameter tanpa nama dalam evaluasi ekspresi, bukan hanya "x", "y", dll.; untuk ini, mulai ekspresi dengan "@(variabel) ...".

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

```
@(a,b) a^2+b^2
41
```

Ini memungkinkan manipulasi ekspresi dalam variabel lain untuk fungsi-fungsi EMT yang membutuhkan ekspresi dalam "x".

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

Seperti yang terlihat pada contoh berikut, variabel global tetap terlihat selama evaluasi.

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

```
-0.475
```

Semua variabel lain dalam ekspresi dapat ditentukan saat evaluasi dengan menggunakan parameter yang diberikan.

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

```
-0.425
```

Sebuah ekspresi tidak harus bersifat simbolik; hal ini diperlukan jika ekspresi tersebut mengandung fungsi yang hanya dikenal di kernel numerik, bukan di Maxima.

Matematika Simbolik

EMT melakukan matematika simbolik dengan bantuan Maxima. Untuk detailnya, mulailah dengan tutorial berikut, atau jelajahi referensi untuk Maxima. Bagi para ahli Maxima perlu dicatat bahwa ada perbedaan dalam sintaks antara sintaks asli Maxima dan sintaks default dari ekspresi simbolik di EMT.

Matematika simbolik terintegrasi mulus ke dalam Euler dengan tanda &. Setiap ekspresi yang diawali dengan & adalah ekspresi simbolik. Ekspresi ini akan dievaluasi dan dicetak oleh Maxima.

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

```
>$&44!
```

$$2658271574788448768043625811014615890319638528000000000$$

Dengan cara ini, Anda dapat menghitung hasil besar secara eksak. Mari kita hitung:

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

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

$$2481256778$$

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

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

$$2481256778$$

Untuk mempelajari lebih lanjut tentang suatu fungsi tertentu, klik ganda pada namanya. Misalnya, coba klik ganda pada &binomial pada baris perintah sebelumnya. Ini akan membuka dokumentasi Maxima sebagaimana disediakan oleh pembuat program itu.

Anda akan mengetahui bahwa perintah berikut juga berfungsi:

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

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

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

Jika Anda ingin mengganti x dengan nilai tertentu gunakan with.

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

$$120$$

Dengan cara ini Anda bisa menggunakan solusi dari suatu persamaan ke dalam persamaan lainnya.

Ekspresi simbolik dicetak oleh Maxima dalam bentuk 2D. Alasannya adalah adanya flag simbolik khusus di dalam string.

Seperti yang sudah terlihat pada contoh sebelumnya, jika Anda sudah menginstal LaTeX, Anda dapat mencetak ekspresi simbolik dengan LaTeX. Jika tidak, perintah berikut akan memberikan pesan kesalahan.

Untuk mencetak ekspresi simbolik dengan LaTeX, gunakan tanda $ di depan & (atau Anda bisa menghilangkan &) sebelum perintah. Jangan jalankan perintah Maxima dengan $, jika Anda tidak punya LaTeX terinstal.

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

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

Ekspresi simbolik diparsing oleh Euler. Jika Anda membutuhkan sintaks kompleks dalam satu ekspresi, Anda bisa menutup ekspresi itu dalam "...". Menggunakan lebih dari satu ekspresi memungkinkan, tetapi sangat tidak disarankan.

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

```
25
```

Sebagai catatan, ekspresi simbolik bisa digunakan dalam program, tapi harus ditulis dalam tanda kutip. Selain itu, jauh lebih efisien memanggil Maxima saat compile time jika memungkinkan.

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

$$x^4+4\,x^3+6\,x^2+4\,x+1$$

$$4\,\left(x+1\right)^3$$

Sekali lagi, % mengacu pada hasil sebelumnya.
Untuk mempermudah, kita bisa menyimpan solusi ke dalam variabel simbolik. Variabel simbolik didefinisikan dengan &=.

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

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

Ekspresi simbolik dapat digunakan dalam ekspresi simbolik lainnya.

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

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

Input langsung perintah Maxima juga tersedia. Mulailah baris perintah dengan ::. Sintaks Maxima disesuaikan dengan sintaks EMT (disebut “compatibility mode”).

```
>&factor(20!)
```

```
2432902008176640000
```

```
>::: factor(10!)
```

```
 8   4   2
2   3   5   7
```

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

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

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

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

```
 2
g
```

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

```
 3   x
x   E
```

$$x^3\,e^x$$

Variabel seperti ini bisa digunakan dalam ekspresi simbolik lainnya. Perhatikan, pada perintah berikut sisi kanan dari &= dievaluasi sebelum penugasan ke fx.

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

```
      5
125  E
```

$$125\,e^5$$

```
18551.64488782208
```

```
>fx(5)
```

```
18551.6448878
```

Untuk mengevaluasi ekspresi dengan nilai tertentu dari variabel, gunakan operator with.

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

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

```
            10         5
     1000 E   - 125 E


   2.20079141499189e+7
```

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

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

Untuk mendapatkan kode LaTeX dari suatu ekspresi, gunakan perintah tex.

```
>tex(fx)
```

```
x^3\,e^{x}
```

Ekspresi simbolik dapat dievaluasi seperti ekspresi numerik.

```
>fx(0.5)
```

```
0.206090158838
```

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

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

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

Penugasan tersebut juga dapat bersifat simbolis.

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

$$\left(t+1\right)^3\,e^{t+1}$$

Perintah solve digunakan untuk menyelesaikan ekspresi simbolik terhadap suatu variabel di Maxima. Hasilnya berupa vektor solusi.

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

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

Sebagai perbandingan, perintah numerik solve di Euler membutuhkan nilai awal, dan opsional target nilai.

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

```
1.56155281281
```

Nilai numerik dari solusi simbolik dapat dihitung dengan mengevaluasi hasil simboliknya. Euler akan membaca penugasan x= ... dan seterusnya. Jika Anda tidak memerlukan nilai numerik untuk perhitungan lebih lanjut, Anda juga bisa membiarkan Maxima menemukan nilai numeriknya.

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

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

```
[-3.23607,  1.23607]
```

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

Untuk mendapatkan solusi simbolik tertentu, Anda bisa menggunakan with 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 simbolik dapat memiliki flags yang menunjukkan perlakuan khusus di Maxima. Beberapa flags dapat digunakan sebagai perintah, lainnya tidak. Flags ditambahkan dengan | (bentuk lain dari ev(...,flags)).

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

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

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

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

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

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

Fungsi

Di EMT, fungsi adalah program yang didefinisikan dengan perintah function. Fungsi bisa berupa satu baris (one-line) atau multi-baris.
Fungsi satu baris bisa numerik atau simbolik. Fungsi numerik satu baris didefinisikan dengan :=.

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

Sebagai gambaran umum, berikut semua kemungkinan definisi untuk fungsi satu baris. Fungsi dapat dievaluasi sama seperti fungsi bawaan Euler.

```
>f(2)
```

```
4.472135955
```

Fungsi ini juga bekerja untuk vektor, mengikuti aturan bahasa matriks di Euler, karena ekspresi yang dipakai di fungsi tersebut ter-vektorisasi.

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

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

Fungsi dapat digambar (plot). Alih-alih ekspresi, kita hanya perlu memberikan nama fungsi.
Sebaliknya dengan ekspresi simbolik atau numerik, nama fungsi harus diberikan dalam bentuk string.

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

```
0.786151377757
```

Secara default, jika Anda ingin menimpa (overwrite) fungsi bawaan, Anda harus menambahkan kata kunci overwrite. Menimpa fungsi bawaan berbahaya karena dapat menyebabkan masalah pada fungsi lain yang bergantung padanya.

Anda masih bisa memanggil fungsi bawaan dengan _..., jika fungsi itu bagian dari inti Euler.

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

```
0.707106781187
```

Sebaiknya kita hapus redefinisi sin ini.

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

```
0.707106781187
```

Parameter Bawaan

Fungsi numerik dapat memiliki parameter bawaan (default).

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

Jika parameter itu tidak diberikan saat pemanggilan, maka nilai bawaan dipakai.

```
>f(4)
```

```
16
```

Jika parameter ditentukan, maka nilai bawaan akan ditimpa.

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

```
80
```

Parameter yang diberikan dengan cara penugasan (assigned parameter) juga akan menimpa nilai bawaan. Hal ini juga digunakan oleh banyak fungsi Euler seperti plot2d, plot3d.

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

```
16
```

Jika variabel bukan merupakan parameter, maka variabel itu harus global. Fungsi satu baris dapat mengakses variabel global.

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

```
24
```

Tetapi parameter yang diberikan dengan penugasan akan mengesampingkan nilai global.

Jika argumen tidak ada di daftar parameter yang sudah ditentukan, maka harus dideklarasikan dengan :=.

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

```
20
```

Fungsi simbolik didefinisikan dengan &=. Fungsi ini didefinisikan baik di Euler maupun di Maxima, sehingga bisa digunakan di keduanya. Ekspresi definisi dijalankan melalui Maxima sebelum didefinisikan.

```
>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
```

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

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

Fungsi simbolik juga bisa digunakan dalam ekspresi numerik. Tentu saja, ini hanya akan berfungsi jika EMT bisa menginterpretasikan semua hal di dalam fungsi tersebut.

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

```
178.635099908
```

Fungsi simbolik bisa dipakai untuk mendefinisikan fungsi simbolik lain atau ekspresi baru.

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

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

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

```
0.703467422498
```

Hal berikut juga berfungsi, karena Euler akan memakai ekspresi simbolik dalam fungsi g, jika tidak menemukan variabel simbolik g, dan jika ada fungsi simbolik g.

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

```
0.703467422498
```

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

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

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

$$(x+2)^n$$

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

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

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

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

```
625
```

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

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

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

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

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

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

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

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

$$(x+2)^3\;(2\,x-1)^4$$

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

$$(x+2)^3\;(2\,x-1)^4$$

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

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

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

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

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

$$x^3-x$$

Dengan &= fungsi menjadi simbolik, dan dapat dipakai dalam ekspresi simbolik lain.

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

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

Dengan := fungsi bersifat numerik. Contoh bagus adalah integral tentu, yang tidak bisa dihitung secara simbolik.

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

Jika kita mendefinisikan fungsi dengan kata kunci map, maka fungsi bisa dipakai untuk vektor x. Secara internal, fungsi dipanggil untuk setiap nilai x satu per satu, 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 bawaan untuk parameter.

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

Sekarang fungsi dapat dipanggil dengan atau tanpa parameter base.

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

```
2
6.7
```

Selain itu, parameter juga bisa diberikan dengan penugasan.

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

```
2
```

Sering kali, kita ingin menggunakan fungsi untuk vektor pada satu konteks, dan untuk elemen individual pada konteks lain. Hal ini dimungkinkan dengan parameter berbentuk 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 ini bisa digunakan untuk variabel simbolik.
etapi fungsi ini juga bisa dipakai untuk vektor numerik.

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

```
17
```

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

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

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

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

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

$$0$$

Tentu saja, fungsi ini bisa digunakan dalam ekspresi simbolik atau dipakai untuk mendefinisikan fungsi simbolik lain.

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

$$10\,\left(y^2+x\right)^3\,\left(9\,y^2+x+2\right)$$

Untuk rangkuman
- &= mendefinisikan fungsi simbolik,
- := mendefinisikan fungsi numerik,
- &&= mendefinisikan fungsi simbolik murni.

Menyelesaikan Ekspresi

Ekspresi dapat diselesaikan baik secara numerik maupun simbolik.

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

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

```
1.41421356237
```

Hal ini juga bisa dipakai untuk ekspresi simbolik. Misalnya:

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

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

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

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

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

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

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

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

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

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

Sekarang kita cari titik di mana polinomial bernilai 2. Dalam solve(), nilai target default adalah y=0, tetapi ini bisa diubah dengan variabel penugasan.

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

```
0.966715594851
2
```

Menyelesaikan ekspresi simbolik dalam bentuk simbolik menghasilkan daftar solusi. Kita gunakan symbolic solver solve() dari Maxima.

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

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

Cara termudah untuk mendapatkan nilai numerik dari solusi adalah mengevaluasi hasilnya secara numerik seperti ekspresi biasa.

```
>longest sol()
```

```
      -0.6180339887498949       1.618033988749895
```

Untuk menggunakan solusi ini secara simbolik dalam ekspresi lain, cara paling mudah adalah dengan with.

```
>$&x^2 with sol[1], $&expand(x^2-x-1 with sol[2])
```

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

$$0$$

Menyelesaikan sistem persamaan secara simbolik dapat dilakukan dengan vektor persamaan dan solve(). Hasilnya berupa daftar solusi dalam bentuk 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() bisa mengakses variabel global. Tetapi sering kali kita ingin menggunakan parameter lokal.

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

Misalnya dengan a=3:

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

Salah satu cara memberikan parameter tambahan ke f() adalah dengan memakai daftar yang berisi nama fungsi dan parameternya (cara lain adalah semicolon parameters).

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

```
2.54116291558
```

Hal ini juga bisa dipakai untuk ekspresi. Tapi kali ini elemen daftar harus diberi nama. (Lebih lanjut tentang daftar ada di tutorial sintaks EMT).

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

```
2.54116291558
```

Menyelesaikan Pertidaksamaan

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

```
>&load(fourier_elim)
```

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

```
>$&fourier_elim([x^2 - 1>0],[x]) // x^2-1 > 0
```

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

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

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

```
>$&fourier_elim([x^2 - 1 # 0],[x]) // x^-1 <> 0
```

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

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

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

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

$$emptyset$$

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

$$universalset$$

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

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

```
>$&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 < - 11]
   or [x < 8, 13 < y] or [x = y, 13 < y] or [8 < x, x < y, 13 < y]
   or [y < x, 13 < y]
```

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

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

Bahasa Matriks di EMT

Dokumentasi inti EMT berisi pembahasan detail tentang bahasa matriks Euler.

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

```
>A=[1,2;3,4]
```

```
              1               2
              3               4
```

Hasil kali matriks ditandai dengan titik.

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

```
              3
              4
```

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

```
  [3,  4]
```

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

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

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

```
             11
             25
```

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

```
              1               0
              0               1
```

Tujuan utama bahasa matriks adalah supaya semua fungsi dan operator bekerja element-wise (per elemen).

```
>A.A
```

```
              7              10
             15              22
```

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

```
              1               4
              9              16
```

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

```
            37            54
            81           118
```

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

```
            37            54
            81           118
```

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

```
             1             1
             1             1
```

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

```
      0.333333      0.666667
          0.75             1
```

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

```
            -2
           2.5
```

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

```
            -2
           2.5
```

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

```
             1             0
             0             1
```

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

```
             1             0
             0             1
```

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

```
             1             4
             9            16
```

Ini bukan hasil perkalian matriks, melainkan perkalian elemen demi elemen. Hal yang sama berlaku untuk vektor.

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

```
              9
             16
```

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

```
>2*A
```

```
              2             4
              6             8
```

Contoh: jika operand adalah vektor kolom, elemennya diterapkan ke semua baris A.

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

```
              1             4
              3             8
```

Jika operand adalah vektor baris, maka diterapkan ke semua kolom A.

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

```
              2             6
              6            12
```

Perkalian vektor baris dan kolom bisa dibayangkan sebagai penggandaan (duplikasi).

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

```
              1             2
              1             2
```

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

```
              1             4
              3             8
```

Hal ini juga berlaku untuk dua vektor, di mana satu adalah vektor baris dan yang lain vektor kolom. Kita bisa menghitung i.j untuk i,j=1..5. Triknya adalah mengalikan 1:5 dengan transposenya.
Bahasa matriks di Euler secara otomatis menghasilkan tabel nilai.

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

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

Ingat, ini bukan 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 banyaknya 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 kesamaan.
Kita mendapat 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 ini, nonzeros akan memilih elemen-elemen tak nol.
Dalam contoh ini, kita mendapatkan indeks dari semua elemen yang lebih besar dari 50.

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

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

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

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

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

Contoh lain: mencari semua kuadrat dari 1 sampai 1000 yang bersisa 5 modulo 11 dan 3 modulo 13.

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

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

EMT tidak sepenuhnya efektif untuk perhitungan bilangan bulat, karena secara internal menggunakan double precision floating point. Namun, sering kali fitur ini tetap berguna.
Kita bisa memeriksa keprimaan suatu bilangan.

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

```
 112
```

Fungsi nonzeros() hanya berlaku untuk vektor. Untuk matriks, gunakan 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
```

Fungsi ini mengembalikan indeks dari elemen-elemen yang memenuhi syarat (tidak nol).

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

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

Indeks-indeks ini bisa digunakan untuk mengganti elemen-elemen dengan suatu 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 bisa dipakai untuk mengisi elemen-elemen pada indeks tertentu dengan nilai dari matriks lain.

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

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

Dan kita juga bisa mengambil elemen-elemen tersebut dalam bentuk vektor.

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

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

Fungsi lain yang berguna adalah extrema, yang mengembalikan nilai minimum dan maksimum dari setiap baris matriks, beserta posisinya.

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

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

Kita bisa menggunakan hasil ini untuk mengambil nilai maksimum dari tiap baris.

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

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

Tentu saja, ini sama saja dengan fungsi max().

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

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

Namun dengan mget(), kita bisa mengekstrak indeks dan menggunakan informasi itu untuk mengambil elemen-elemen dari matriks lain pada posisi yang sama.

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

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

Membangun Matriks

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

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

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

Demikian juga, kita bisa menyusun matriks secara berdampingan, asalkan jumlah barisnya sama.

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

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

Jika jumlah barisnya tidak sama, maka matriks yang lebih pendek akan diisi dengan 0.

da pengecualian untuk aturan ini: sebuah bilangan real yang ditempelkan ke sebuah matriks akan diperlakukan sebagai kolom yang seluruh elemennya berisi bilangan tersebut.

```
>A|1
```

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

Kita juga dapat membuat sebuah matriks dari vektor baris maupun vektor kolom.

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

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

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

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

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

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

```
           1             1
           2             4
           3             9
```

Untuk mendapatkan ukuran dari sebuah matriks A, kita dapat menggunakan fungsi berikut:

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

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

Untuk vektor, gunakan length().

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

```
9
```

Ada banyak fungsi lain yang bisa digunakan untuk menghasilkan matriks.

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

```
            1             1
            1             1
```

Fungsi ini juga dapat digunakan dengan satu parameter. Untuk mendapatkan vektor dengan nilai selain 1, gunakan cara berikut:

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

```
  [6,  6,  6,  6,  6]
```

Selain itu, matriks bilangan acak dapat dihasilkan dengan random (distribusi uniform) atau normal (distribusi Gauss).

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

```
      0.66566      0.831835
        0.977      0.544258
```

Fungsi lain yang berguna adalah redim(), yang menyusun ulang elemen-elemen sebuah matriks menjadi bentuk matriks baru.

```
>redim(1:9,3,3) // menyusun elemen2 1, 2, 3, ..., 9 ke bentuk matriks 3x3
```

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

Dengan fungsi ini serta fungsi dup(), kita dapat menuliskan fungsi rep(), yang mengulang sebuah vektor sebanyak n kali.

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

Mari kita uji.

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

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

Fungsi multdup() menggandakan elemen-elemen dari sebuah vektor.

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

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

Fungsi flipx() dan flipy() membalik urutan baris atau kolom dari sebuah matriks. Misalnya, flipx() membalik secara horizontal.

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

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

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

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

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

Fungsi khusus lain adalah drop(v,i), yang menghapus elemen-elemen dengan indeks i dari vektor v.

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

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

Perlu dicatat bahwa vektor i dalam drop(v,i) mengacu pada indeks elemen dalam v, bukan nilai elemennya. Jika ingin menghapus elemen tertentu, elemen tersebut harus dicari terlebih dahulu. Fungsi indexof(v,x) bisa digunakan untuk menemukan elemen x dalam vektor v yang terurut.

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

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

Tidak masalah jika indeks yang dimasukkan di luar jangkauan, berulang, atau tidak terurut.

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

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

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

Kita mulai dengan matriks identitas.

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

```
            1            0            0            0            0
            0            1            0            0            0
            0            0            1            0            0
            0            0            0            1            0
            0            0            0            0            1
```

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

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

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

Perhatikan bahwa kita tidak mengubah matriks A. Kita mendapatkan matriks baru sebagai hasil dari setdiag().

Berikut adalah sebuah fungsi yang menghasilkan matriks tridiagonal.

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

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

Diagonal dari sebuah matriks juga bisa diekstrak. Untuk menunjukkannya, kita ubah vektor 1:9 menjadi matriks 3x3.

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

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

Sekarang kita dapat mengekstrak diagonalnya.

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

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

Sebagai contoh, kita bisa membagi matriks dengan diagonalnya. Bahasa matriks akan memastikan bahwa vektor kolom d diterapkan pada matriks tersebut baris demi baris.

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

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

Vektorisasi

Hampir semua fungsi di Euler juga berlaku untuk input berupa matriks atau vektor, selama hal itu masuk akal.

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

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

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

Dengan ini, kita bisa dengan mudah membuat tabel nilai. Ini juga salah satu cara untuk memplot sebuah fungsi (alternatif lain adalah menggunakan ekspresi).

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

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

Dalam contoh berikut, kita menghasilkan vektor nilai t[i] dengan jarak 0,1 dari -1 sampai 1. Lalu kita menghasilkan vektor nilai fungsi s = t³ - t.

$$s = t^3 - t$$

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

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

Operator di EMT diperluas untuk skalar, vektor, dan matriks dengan cara yang jelas.

Misalnya, sebuah vektor kolom dikalikan dengan vektor baris akan menghasilkan matriks, jika suatu operator diterapkan.

```
>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
```

Perlu dicatat bahwa ini berbeda dengan hasil kali matriks. Hasil kali matriks di EMT ditulis dengan tanda titik ".".

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

```
55
```

Secara bawaan, vektor baris ditampilkan dalam format ringkas.

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

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

Untuk matriks, operator khusus "." digunakan untuk perkalian matriks, dan tanda petik ′ digunakan untuk transpose. Sebuah matriks 1x1 bisa diperlakukan sama seperti bilangan real.

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

```
5
25
```

Transpose matriks dilakukan dengan tanda petik ′.

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

```
            1
            2
            3
            4
```

So we can compute matrix A times vector b.

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

```
           30
           70
```

Dengan ini, kita dapat menghitung hasil kali matriks A dengan vektor b.

```
>v'.v
```

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

Perlu dicatat bahwa v masih berupa vektor baris. Jadi v′·v berbeda dengan v·v′. Hasil v·v′ adalah norma kuadrat dari vektor baris v. Hasilnya berupa vektor 1x1, yang bisa diperlakukan sama seperti bilangan real.

```
>v.v'
```

```
30
```

Selain itu, ada fungsi norm() (bersama dengan banyak fungsi aljabar linear lainnya).

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

```
30
```

Operator dan fungsi mengikuti bahasa matriks Euler.
ingkasnya:
-Sebuah fungsi yang diterapkan pada vektor atau matriks akan diterapkan pada tiap elemennya.

-Operator yang bekerja pada dua matriks berukuran sama akan diterapkan pasangan demi pasangan pada setiap elemen.

-Jika dua matriks memiliki dimensi berbeda, keduanya akan diperluas secara wajar agar ukurannya sama.

Sebagai contoh:

-Sebuah skalar dikalikan dengan vektor maka nilainya dikalikan ke setiap elemen vektor.

-Sebuah matriks dikalikan dengan vektor (dengan *, bukan ".") maka vektor diperluas ke ukuran matriks dengan cara diduplikasi.
Contoh sederhana adalah operator pangkat ^.

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

```
[1,  4,  9]
```

Kasus yang lebih rumit adalah vektor baris dikalikan dengan vektor kolom, keduanya diperluas dengan duplikasi.

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

```
            1             2             3
            2             4             6
            3             6             9
```

Catatan penting: hasil kali skalar menggunakan perkalian matriks ".", bukan *.

```
>v.v'
```

```
14
```

Ada banyak sekali fungsi untuk matriks. Dokumentasi memberikan daftar lengkap, tapi beberapa di antaranya adalah:

-sum, prod = menghitung jumlah dan hasil perkalian per baris
-cumsum, cumprod = melakukan hal yang sama secara kumulatif
-extrema = mengembalikan nilai minimum dan maksimum tiap baris beserta posisinya
-diag(A,i) = mengembalikan diagonal ke-i
-setdiag(A,i,v) = mengganti diagonal ke-i dengan v
-id(n) = matriks identitas
-det(A) = determinan
-charpoly(A) = polinomial karakteristik
-eigenvalues(A) = nilai eigen

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

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

Operator titik dua : menghasilkan vektor baris dengan jarak yang sama, opsional dengan langkah tertentu.

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

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

Untuk menggabungkan matriks dan vektor, gunakan operator "|" (samping) dan "_" (atas-bawah).

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

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

Elemen-elemen matriks bisa diakses 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] mengembalikan elemen ke-i. Untuk matriks, ini mengembalikan baris ke-i.

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

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

Indeks juga bisa berupa vektor indeks. Simbol : berarti semua indeks. Bentuk pendeknya adalah dengan mengosongkan 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 keperluan vektorisasi, elemen-elemen matriks bisa diakses seakan-akan matriks adalah vektor satu dimensi.

```
>A{4}
```

```
4
```

Matriks juga bisa “diratakan” menggunakan fungsi redim(). Ada juga fungsi khusus flatten() untuk ini.

```
>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 sebagai tabel, mari kita atur ulang format ke pengaturan awal, lalu hitung tabel nilai sinus dan kosinus. Perlu diperhatikan bahwa sudut-sudut dihitung dalam satuan radian secara bawaan.

```
>defformat; w=0°:45°:360°; w=w'; deg(w)
```

```
             0
            45
            90
           135
           180
           225
           270
           315
           360
```

Sekarang kita tambahkan kolom-kolom ke sebuah matriks.
Kode Akhir:

```
>M = deg(w)|w|cos(w)|sin(w)
```

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

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

Dalam contoh berikut, kita menghitung t[j]^i untuk i dari 1 sampai n. Hasilnya adalah sebuah matriks, di mana setiap baris berisi tabel t^i untuk satu nilai i. Dengan kata lain, matriks ini memiliki elemen:

a(i,j) = t[j]^i, dengan 1 = j = 101 dan 1 = i = n.

Jika sebuah fungsi tidak bisa menerima input berupa vektor, fungsi itu harus “divektorisasi”.

Hal ini bisa dilakukan dengan menambahkan kata kunci map pada definisi fungsi. Dengan cara ini, fungsi akan dievaluasi untuk setiap elemen dari parameter vektor.

Integrasi numerik integrate() hanya bisa digunakan untuk batas interval skalar. Oleh karena itu, kita perlu membuatnya bisa dipakai untuk vektor dengan memanfaatkan kata kunci map.

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

Kata kunci map melakukan vektorisasi terhadap fungsi. Dengan begitu, fungsi tersebut dapat bekerja pada vektor bilangan.

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

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

Sub-Matriks dan Element Matriks

Untuk mengakses sebuah elemen matriks, gunakan notasi tanda kurung siku.

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

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

Kita bisa mengakses satu baris penuh dari sebuah matriks.

```
>A[2]
```

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

Jika berupa vektor baris atau kolom, ini akan mengembalikan satu elemen dari vektor tersebut.

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

```
2
```

Untuk memastikan bahwa kita mendapatkan baris pertama baik pada matriks berukuran 1×n maupun m×n, kita harus menuliskan semua kolom dengan indeks kedua dikosongkan.

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

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

Jika indeks berupa vektor indeks, Euler akan mengembalikan baris-baris dari matriks yang sesuai.
Sebagai contoh, di sini kita ingin mengambil baris pertama dan kedua dari A.

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

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

Kita bahkan bisa menyusun ulang A menggunakan vektor indeks.
erlu dicatat bahwa dalam hal ini kita tidak mengubah A, melainkan membuat versi A yang sudah disusun ulang.

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

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

Trik indeks ini juga berlaku untuk kolom.

Sebagai contoh, kita memilih semua baris dari A dan hanya kolom ke-2 serta ke-3.

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

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

Sebagai singkatan, tanda titik dua ":" digunakan untuk mewakili semua indeks baris atau kolom.

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

```
3
6
9
```

Sebagai alternatif, kita bisa membiarkan indeks pertama kosong.

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

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

Kita juga bisa mengambil baris terakhir dari A.

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

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

Sekarang mari kita ubah elemen-elemen dari A dengan menetapkan nilai pada sebuah sub-matriks dari A. Berbeda dengan contoh sebelumnya, kali ini matriks A yang tersimpan memang benar-benar berubah.

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

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

Kita juga bisa memberikan nilai pada seluruh baris dalam A.

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

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

Bahkan, kita bisa menetapkan nilai pada sebuah sub-matriks asalkan ukurannya sesuai.

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

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

Selain itu, ada beberapa cara pintas yang diperbolehkan.

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

```
              0             0            -1
              0             0             6
              7             8             9
```

Peringatan: Indeks yang berada di luar batas akan mengembalikan matriks kosong atau pesan kesalahan, tergantung pada pengaturan sistem. Secara bawaan, hasilnya adalah pesan kesalahan. Namun, perlu diingat bahwa indeks negatif dapat digunakan untuk mengakses elemen matriks dengan menghitung dari bagian akhir.

```
>A[4]
```

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

Pengurutan dan Pengacakan

Fungsi sort() mengurutkan sebuah vektor baris.

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

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

Sering kali, kita juga perlu mengetahui indeks dari vektor yang sudah diurutkan terhadap posisi aslinya. Indeks ini bisa digunakan untuk menyusun ulang vektor lain dengan cara yang sama.

Mari kita coba mengacak sebuah vektor.

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

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

Indeks hasil pengurutan akan berisi urutan yang benar dari elemen-elemen vektor v. Dengan begitu, kita bisa menggunakan indeks ini untuk menempatkan kembali elemen-elemen v dalam urutan yang benar.

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

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

Fungsi ini juga berfungsi untuk vektor string.

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

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

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

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

Seperti yang bisa kamu lihat, posisi elemen ganda (duplikat) agak acak.

```
>ind
```

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

Fungsi unique mengembalikan daftar elemen unik dari sebuah vektor, dalam keadaan sudah terurut.

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

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

Fungsi ini juga bisa digunakan untuk vektor string.

```
>unique(s)
```

```
a
aa
d
e
```

Aljabar Linear

EMT memiliki banyak fungsi untuk menyelesaikan sistem linear, sistem jarang (sparse systems), atau masalah regresi.

Untuk sistem linear Ax = b, kita bisa menggunakan algoritma Gauss, matriks invers, atau linear fit.
perator A\b menggunakan versi dari algoritma Gauss.

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

```
            -4
           4.5
```

Dalam contoh, matriks A berukuran 2×2 dan vektor b diberikan. Hasil A\b adalah solusi dari sistem persamaan tersebut.
Contoh lain: kita membuat matriks 200×200 dan menghitung jumlah tiap barisnya. Kemudian kita menyelesaikan Ax = b menggunakan matriks invers. Kesalahan diukur sebagai deviasi maksimum semua elemen dari 1 (karena solusi yang benar seharusnya vektor berisi 1). Hasilnya sangat kecil, mendekati nol, sehingga metode ini akurat.

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

```
    8.790745908981989e-13
```

Jika sistem tidak memiliki solusi, maka linear fit akan meminimalkan norma dari kesalahan Ax - b.

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

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

Determinannya dari matriks ini adalah 0.

```
>det(A)
```

```
0
```

Matriks Simbolik

Maxima memiliki kemampuan untuk menangani matriks simbolik.
entu saja, Maxima juga bisa digunakan untuk menyelesaikan persoalan aljabar linear sederhana.Kita dapat mendefinisikan matriks untuk Euler dan Maxima dengan operator &:=, lalu menggunakannya dalam ekspresi simbolik.Bentuk penulisan matriks biasa dengan tanda [...] juga 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(%)
```

$$a\,\left(a^2-1\right)-2\,a+2$$

$$\left(a-1\right)^2\,\left(a+2\right)$$

```
>$&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 halnya variabel simbolik lain, matriks simbolik juga bisa digunakan dalam ekspresi simbolik.

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

$$\left(1-x\right)\,\left(2-x\right)-a\,b$$

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

Nilai eigen juga dapat dihitung secara otomatis. Hasilnya berupa sebuah vektor yang berisi dua bagian: nilai eigen dan kelipatannya.

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

$$\left[\left[a-1 , a+1\right] , \left[1 , 1\right]\right]$$

Untuk mengekstrak sebuah vektor eigen tertentu diperlukan pemanggilan indeks yang hati-hati.

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

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

```
[1, - 1]
```

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

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

```
              1              4
              5              2
```

Dalam ekspresi simbolik, kita bisa menggunakan kata kunci with.

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

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

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

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

$$[1,a]$$

Sebuah ekspresi simbolik bisa berisi penugasan (assignment). Penugasan ini akan mengubah matriks A.

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

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

Di Maxima juga tersedia fungsi-fungsi simbolik untuk membuat vektor dan matriks.
ntuk hal ini, silakan merujuk pada 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)
```

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

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

Hasil dari ekspresi simbolik dapat dievaluasi secara numerik di Euler.
ntuk informasi lebih lanjut tentang Maxima, lihat pengantar Maxima.

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

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

Euler juga memiliki fungsi yang kuat yaitu xinv(), yang berusaha lebih keras dan memberikan hasil yang lebih akurat.

Perlu diperhatikan bahwa dengan &:=, matriks B telah didefinisikan sebagai matriks simbolik untuk ekspresi simbolik, dan sekaligus sebagai matriks numerik untuk ekspresi numerik. Jadi kita bisa menggunakannya dalam kedua konteks.

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

```
                        1                       0
                        0                       1
```

Sebagai contoh, kita bisa menghitung nilai eigen dari matriks A secara numerik.

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

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

Atau secara simbolik. Untuk detailnya, lihat tutorial tentang Maxima.

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

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

Nilai Numerik dan Ekspresi Simbolik

Ekspresi simbolik hanyalah sebuah string yang berisi ekspresi.
ika kita ingin mendefinisikan sebuah nilai sekaligus untuk ekspresi simbolik dan ekspresi numerik, kita harus menggunakan &:=.

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

```
                  1         3.14159
                  4               5
```

Masih ada perbedaan antara bentuk numerik dan bentuk simbolik.
aat mentransfer matriks ke bentuk simbolik, nilai riil akan diubah menjadi bentuk pecahan atau pendekatan pecahan.

```
>$&A
```

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

Untuk menghindari hal ini, tersedia fungsi mxmset(variable).

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

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

Maxima juga bisa melakukan perhitungan dengan bilangan pecahan desimal (floating point numbers), bahkan dengan bilangan pecahan besar (big floating numbers) hingga 32 digit. Namun, evaluasinya akan jauh lebih lambat.

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

$$1.4142135623730950488016887242096980785696718753769480731766797379907324784621070388503875343276415 73_B \times 10^0$$

$$1.414213562373095$$

Presisi dari bilangan pecahan desimal besar (big floating point numbers) dapat diubah.

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

```
          3.14159265358979323846264338327950288419716939937510582097494\
  4592307816406286208998628034825342117068b0
```

Sebuah variabel numerik dapat digunakan di dalam ekspresi simbolik dengan menuliskannya menggunakan "@var".

Perlu dicatat bahwa hal ini hanya diperlukan jika variabel tersebut telah didefinisikan dengan := atau = sebagai variabel numerik.

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

$$-5.424777960769379$$

Demo - Suku Bunga

Di bawah ini, kita menggunakan Euler Math Toolbox (EMT) untuk perhitungan bunga.
ita lakukan baik secara numerik maupun simbolik untuk menunjukkan bagaimana Euler bisa digunakan dalam menyelesaikan masalah nyata.

Misalkan kita memiliki modal awal sebesar 5000 (misalnya dalam dolar).

```
>K=5000
```

```
5000
```

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

```
>K*1.03
```

```
5150
```

Euler juga bisa memahami sintaks seperti.

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

```
5150
```

Namun, lebih mudah jika kita menggunakan faktor.

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

```
1.03
5150
```

Untuk jangka 10 tahun, kita cukup mengalikan faktor tersebut berulang kali, sehingga kita mendapatkan nilai akhir dengan bunga berbunga (compound interest).

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

```
6719.58189672
```

Untuk keperluan ini, kita bisa mengatur format hasil perhitungan menjadi 2 digit setelah tanda desimal.

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

```
     6719.58
```

Kita juga bisa mencetak hasil yang sudah dibulatkan ke 2 digit dalam sebuah kalimat lengkap.

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

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

Bagaimana kalau kita ingin mengetahui hasil di setiap tahun, dari tahun ke-1 sampai ke-9?
i sini, bahasa matriks Euler sangat membantu.
ita tidak perlu menuliskan loop, cukup memasukkan:

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

```
Real 1 x 11 matrix

    5000.00     5150.00     5304.50     5463.64     ...
```

Bagaimana cara kerja "keajaiban" ini?
Pertama, ekspresi 0:10 menghasilkan vektor bilangan bulat dari 0 sampai 10.

```
>short 0:10
```

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

Semua operator dan fungsi di Euler dapat diterapkan ke vektor, elemen demi elemen.

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

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

adalah vektor faktor dari q^0 sampai q^10.
etika dikalikan dengan K, kita mendapatkan vektor nilai bunga dari tahun ke-0 sampai tahun ke-10.

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

Namun, cara yang lebih realistis untuk menghitung bunga adalah dengan membulatkan ke sen (cent) terdekat setelah setiap tahun. Mari kita tambahkan fungsi untuk melakukan hal ini:

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

Kemudian kita bisa membandingkan hasil dengan pembulatan dan tanpa pembulatan.

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

```
                  1271.61
                1271.6071
```

Dengan adanya pembulatan, tidak ada lagi rumus sederhana untuk nilai pada tahun ke-n. Maka kita harus menghitungnya tahun demi tahun.

Euler menyediakan beberapa cara untuk hal ini. Cara yang paling mudah adalah menggunakan fungsi iterate, yang akan mengulang sebuah fungsi tertentu sebanyak jumlah iterasi yang diminta.

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

```
  Real 1 x 11 matrix

      5000.00     5150.00     5304.50     5463.64     ...
```

Kita kemudian dapat mencetak hasil tersebut dengan format angka tetap, misalnya dengan dua digit desimal.

```
>VKr'
```

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

Untuk mengambil elemen tertentu dari sebuah vektor, kita gunakan indeks di dalam tanda kurung siku.

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

```
      5150.00
      5000.00     5150.00     5304.50
```

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

Sekarang, mari kita bandingkan elemen terakhir dari hasil pembulatan (VKr) dengan hasil perhitungan penuh.

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

```
      6719.60
      6719.58
```

Perbedaannya sangat kecil

Menyelesaikan Persamaan

Sekarang kita ambil fungsi yang sedikit lebih kompleks, yaitu fungsi yang menambahkan sejumlah uang tertentu setiap tahun.

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

Kita tidak perlu menentukan nilai q atau R saat mendefinisikan fungsi. Nilai-nilai itu hanya perlu didefinisikan ketika fungsi dijalankan. Misalnya, dengan R = 200, hasil iterasi menunjukkan bahwa modal bertambah setiap tahun.

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

```
  Real 1 x 11 matrix

      5000.00      5350.00      5710.50      6081.82     ...
```

"Bagaimana jika kita menarik jumlah yang sama setiap tahun?"

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

```
  Real 1 x 11 matrix

      5000.00      4950.00      4898.50      4845.45     ...
```

Hasil iterasi menunjukkan bahwa modal berkurang dari tahun ke tahun.
Kita bisa melihat, ketika bunga di tahun pertama hanya menghasilkan 150, tetapi kita menarik 200, maka modal berkurang setiap tahunnya.

Bagaimana cara menentukan berapa tahun modal itu akan bertahan?
alah satu cara adalah dengan menuliskan loop per tahun.
amun cara paling mudah adalah menjalankan iterasi cukup panjang.

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

```
  Real 1 x 51 matrix

      5000.00      4950.00      4898.50      4845.45     ...
```

Dengan bahasa matriks Euler, kita bisa menentukan kapan pertama kali nilai modal menjadi negatif dengan cara memeriksa hasil iterasi.

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

```
          48.00
```

Alasannya adalah karena nonzeros(VKR<0) mengembalikan vektor indeks i di mana VKR[i] < 0, dan kemudian fungsi min mengambil indeks terkecil.

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

Fungsi iterate() punya trik tambahan: ia bisa menerima syarat akhir (end condition) sebagai argumen. Dalam kasus itu, hasil yang dikembalikan adalah nilai akhir serta jumlah iterasi.

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

```
         -19.83
          47.00
```

Sekarang mari kita coba menjawab pertanyaan yang lebih ambigu.
isalkan kita tahu bahwa setelah 50 tahun nilainya menjadi 0. Berapa tingkat bunga tahunan yang sesuai?

Ini adalah pertanyaan yang hanya bisa dijawab secara numerik.
i bawah ini, kita akan menurunkan rumus yang diperlukan. Namun bisa dilihat bahwa tidak ada rumus sederhana untuk tingkat bunga. Untuk sementara, kita mengincar solusi numerik.

Langkah pertama adalah mendefinisikan sebuah fungsi yang melakukan iterasi sebanyak n kali. Kita sertakan semua parameter ke dalam fungsi ini.

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

Iterasinya sama seperti sebelumnya:

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

Namun, kali ini kita tidak lagi menggunakan nilai global R dalam ekspresi.
ungsi seperti iterate() di Euler memiliki trik khusus: kita bisa mengoper nilai variabel dalam ekspresi sebagai parameter tambahan menggunakan titik koma (;). Dalam kasus ini, parameter yang diberikan adalah P dan R.

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

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

```
-19.83
```

Sekarang kita bisa menyelesaikan persoalan kita.

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

```
3.15
```

Artinya, tingkat bunga yang sesuai adalah 3.15% per tahun. Kita menggunakan nilai awal 3% sebagai tebakan untuk algoritma. Fungsi solve() memang selalu memerlukan nilai awal.

Dengan fungsi yang sama, kita bisa menjawab pertanyaan berikut:
Berapa jumlah yang bisa kita tarik setiap tahun agar modal awal habis dalam 20 tahun, dengan asumsi tingkat bunga 3% per tahun?

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

```
-336.08
```

Perlu dicatat bahwa kita tidak bisa menyelesaikan masalah untuk variabel jumlah tahun, karena fungsi kita mengasumsikan n sebagai bilangan bulat.

Solusi Simbolik untuk Masalah Suku Bunga

Kita bisa menggunakan bagian simbolik dari Euler untuk mempelajari masalah ini. Pertama, kita definisikan fungsi onepay() secara simbolik.

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

$$R+q\,K$$

Kemudian, kita lakukan iterasi fungsi ini beberapa kali.

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

$$q\,\left(q\,\left(q\,\left(R+q\,K\right)+R\right)+R\right)+R$$

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

Hasil ekspansi menunjukkan sebuah pola.
Setelah n periode, kita memiliki:

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

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

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

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

Dengan menggunakan perintah sum di Maxima, kita bisa menuliskan jumlah deret tersebut.
Agar hasilnya disederhanakan ke bentuk pecahan, kita perlu menambahkan bendera (flag) simpsum.
Selanjutnya, kita bisa membuat sebuah fungsi untuk menuliskan hasil umum dari perhitungan ini.

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

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

Fungsi simbolik yang kita buat bekerja sama seperti fungsi f sebelumnya, tetapi lebih efisien.

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

```
-19.82504734650985
-19.82504734652684
```

Kita bisa menggunakannya untuk mencari waktu n, yaitu kapan modal habis.
engan tebakan awal 30 tahun.

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

```
20.51
```

Artinya, modal akan menjadi negatif setelah 21 tahun.

Kita juga bisa menggunakan sisi simbolik Euler untuk menghitung rumus pembayaran (payments).

Misalkan kita mendapat pinjaman sebesar K, lalu membayar n kali cicilan sebesar R (dimulai setelah tahun pertama), dan menyisakan sisa utang sebesar K? pada saat pembayaran terakhir.
umus untuk ini adalah:

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

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

Biasanya, rumus ini dituliskan dalam bentuk

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

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

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

Kita bisa menyelesaikan persamaan secara simbolik untuk R.

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

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

Namun, dari rumus tersebut terlihat bahwa jika i = 0 (tidak ada bunga), fungsi akan menghasilkan kesalahan desimal (floating point error). Meskipun demikian, Euler tetap dapat memplot hasilnya.

Tentu saja, kita bisa menghitung limitnya.

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

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

Tanpa bunga (i = 0), kita harus membayar kembali 10 kali cicilan sebesar 500.

Persamaan ini juga bisa diselesaikan untuk n. Hasilnya akan lebih jelas jika kita melakukan beberapa penyederhanaan pada ekspresinya.

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

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

ALJABAR

PERSAMAAN DAN PERTIDAKSAMAAN LINEAR

Unsur-Unsur Aljabar Bentuk aljabar adalah representasi matematis

yang menggabungkan angka dan huruf, yang didalamnya terdapat beberapa unsur kunci, yaitu:
a. Variabel
b. Koefisien
c. Konstanta
d. Suku

Contoh:

$$3x+9$$

- Angka 3 disebut sebagai koefisien
- x disebut sebagai variabel
- Angka 9 disebut sebagai
- 3x disebut sebagai suku

Persamaan

Sebuah persamaan didefinisikan sebagai kalimat terbuka yang dihubungkan dengan tanda sama dengan (=). Tujuan utama dari menyelesaikan persamaan adalah untuk menemukan nilai variabel yang membuat kalimat terbuka tersebut menjadi pernyataan yang bernilai benar.

1. Persamaan Linear

$$ax+by=c$$

2. Persamaan Kuadrat

$$ax^2+bx+c=0$$

3. Persamaan Nilai Mutlak

$$|2x-7|=5$$

Contoh

1. Tentukan nilai x dari persamaan berikut.

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

```
>$solve(4*(x-1)=3*(2*x+1), x)
```

$$\left[x = -\frac{7}{2}\right]$$

EMT menyediakan penyelesaian simbolik ($solve()) yang merupakan metode paling kuat untuk mencari solusi eksak. Perintah $solve() dari Maxima digunakan untuk menemukan akar-akar analitik dari sebuah persamaan atau sistem persamaan. Sistem dasarnya adalah:
1. &solve('persamaan', 'variabel') untuk persamaan tunggal
2. $solve([eq1, eq2], [var1, var2]) untuk sistem persamaan

2. Tentukan akar-akar persamaan dari persamaan kuadrat berikut.

$$x^2 - 5x + 6 = 0$$

```
>$solve(x^2-5*x+6=0, x)
```

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

PERTIDAKSAMAAN

Pertidaksamaan adalah kalimat terbuka yang menunjukkan perbandingan ukuran dua objek atau lebih menggunakan tanda ketidaksamaan.

1. Pertidaksamaan Linear

$$2x - 1 > 3$$

2. Pertidaksamaan Kuadrat

$$x^2 - 5x - 6 > 0$$

3. Pertidaksamaan Nilai Mutlak

$$|2x - 5| < 9$$

1. Tentukan penyelesaian dari pertidaksamaan kuadrat berikut.

$$x^2 - 5x - 6 > 0$$

```
>&load(fourier_elim)
```

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

```
>$&fourier_elim([x^2-5*x-6>0], [x])
```

$$[6 < x] \vee [x < -1]$$

Untuk menyelesaikan pertidaksamaan secara simbolik, EMT memanfaatkan paket "fourlier_elim" dari Maxima dengan dua langkah penting yang harus dilakukan, yaitu:
1. Memuat Paket
Gunakan perintah &load(fourier_elim) untuk memuat paket yang diperlukan dari Maxima.
2. Menjalankan Paket
Gunakan sintaks $&fourier_elim([pertidaksamaan],[variabel]) untuk menyelesaikan pertidaksamaan.

TRIGONOMETRI

Trigonometri adalah salah satu cabang matematika yang sangat fundamental dan memiliki peran penting dalam berbagai bidang kehidupan. Kata "trigonometri" berasal dari bahasa Yunani, yaitu "trigonon" yang berarti segitiga dan "metron" yang berarti pengukuran. Secara harfiah, trigonometri dapat diartikan sebagai ilmu yang mempelajari pengukuran segitiga, khususnya hubungan antara sudut dan sisi-sisi dalam segitiga.

Fungsi Trigonometri Dasar:
1. Sinus

$$sin = depan/miring$$

2. Cosinus

$$con = samping/miring$$

3. Tangen

$$depan/samping$$

Contoh Perhitungan

```
>sin (30*pi/180)
```

```
0.50
```

Identitas Trigonometri

Identitas trigonometri adalah persamaan yang melibatkan fungsi-fungsi trigonometri dan berlaku untuk semua nilai sudut yang memenuhi domain fungsi tersebut. Identitas ini tidak bergantung pada nilai spesifik sudut, melainkan merupakan kebenaran umum yang digunakan untuk menyederhanakan perhitungan atau membuktikan hubungan antar fungsi trigonometri.

```
>&trigsimp (sin(x)^2+cos(x)^2)
```

```
1
```

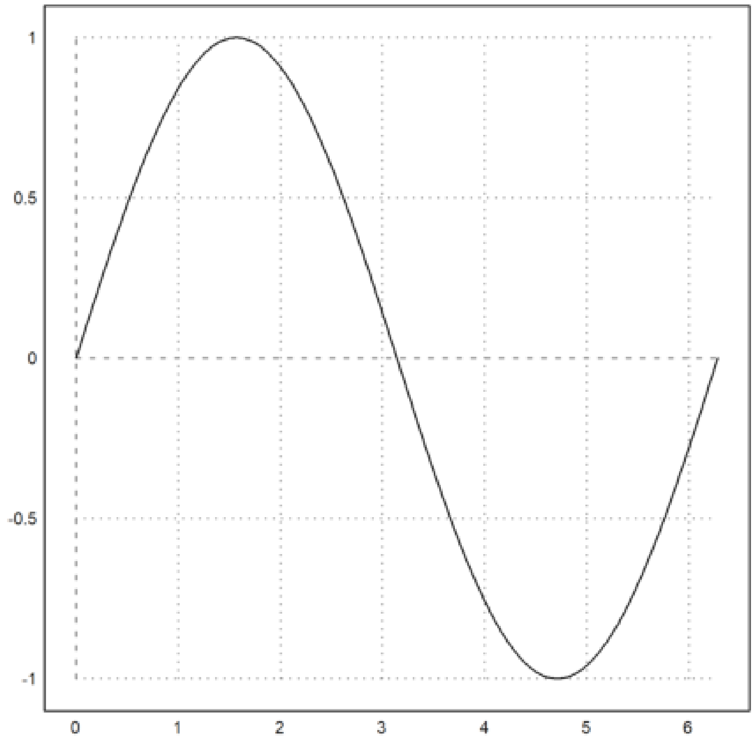
```
>&trigexpand(cos(a+b))
```

```
cos(a) cos(b) - sin(a) sin(b)
```

Grafik Fungsi Trigonometri

Grafik fungsi trigonometri adalah representasi visual dari fungsi-fungsi trigonometri

```
>plot2d("sin(x)", 0, 2*pi):
```



MATRIKS

Matriks adalah array (daftar) bilangan yang terdiri dari baris-baris dan kolom-kolom yang ditulis dalam tanda kurung. Matriks yang biasa kita jumpai adalah matriks yang susunan elemennya berbentuk persegi panjang. Matriks dilambangkan dengan huruf besar.

Jenis-Jenis Matriks

1. Matriks nol

```
>$&Z=[0,0,0;0,0,0]
```

$$Z = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

2. Matriks Baris

```
>$&M=[1,-5,8]
```

$$M = [1, -5, 8]$$

3. Matriks Kolom

```
>$&C=[9;8]
```

$$C = \begin{pmatrix} 9 \\ 8 \end{pmatrix}$$

4. Matriks Persegi

```
>$&D=[1,2;3,4]
```

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

5. Matriks Diagonal

```
>$&F=[5,0,0;0,6,0;0,0,7]
```

$$F = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 7 \end{pmatrix}$$

6. Matriks Segitiga Atas

```
>$&G=[7,8,9;0,1,2;0,0,6]
```

$$G = \begin{pmatrix} 7 & 8 & 9 \\ 0 & 1 & 2 \\ 0 & 0 & 6 \end{pmatrix}$$

7. Matriks Segitiga Bawah

```
>$&H=[3,0,0;4,5,0;1,2,6]
```

$$H = \begin{pmatrix} 3 & 0 & 0 \\ 4 & 5 & 0 \\ 1 & 2 & 6 \end{pmatrix}$$

8. Matriks Identitas

```
>$&J=[1,0,0;0,1,0;0,0,1]
```

$$J = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Contoh Penyelesaian

1. Tentukan penjumlahan dan pengurangan matriks berikut.

```
>$&N=[1,-3;-2,7]
```

$$N = \begin{pmatrix} 1 & -3 \\ -2 & 7 \end{pmatrix}$$

```
>N=[1,-3;-2,7]
```

```
            1.00          -3.00
           -2.00           7.00
```

```
>$&O=[7,3;2,1]
```

$$O = \begin{pmatrix} 7 & 3 \\ 2 & 1 \end{pmatrix}$$

```
>O=[7,3;2,1]
```

```
            7.00           3.00
            2.00           1.00
```

```
>N+O
```

```
            8.00           0.00
            0.00           8.00
```

```
>N-O
```

```
           -6.00          -6.00
           -4.00           6.00
```

2. Tentukan determinan dan invers dari matriks berikut.

```
>$&Y=[-2,5,3;4,-1,3;7,-2,5]
```

$$Y = \begin{pmatrix} -2 & 5 & 3 \\ 4 & -1 & 3 \\ 7 & -2 & 5 \end{pmatrix}$$

```
>Y=[-2,5,3;4,-1,3;7,-2,5]
```

```
    -2.00         5.00         3.00
     4.00        -1.00         3.00
     7.00        -2.00         5.00
```

```
>det(Y)
```

```
     0.00
```

```
>inv(Y)
```

```
Determinant zero!
Try "trace errors" to inspect local variables after errors.
inv:
    return A\id(cols(A));
Error in:
inv(Y) ...
      ^
```

Matriks A tidak mempunyai invers karena determinannya adalah nol. Oleh karena itu, EMT menampilkan output error.

FUNGSI KUADRAT

Fungsi kuadrat merupakan sebuah fungsi polinomial atau sukku banyak dimana pangkat tertinggi dari variabel atau peubahnnya adalah 2. Bentuk umum dari persamaan kuadrat adalah sebagai berikut.

$$ax^2 + bx + c = 0$$

Sedangkan untuk rumus persamaan fungsi kuadrat adalah sebagai berikut.

$$f(x) = ax^2 + bx + c$$

Keterangan:
x = Variabel
a, b = Koefisien
c = Kosntanta

- Salah satu syarat dari suatu fungsi kuadrat adalah bahwa koefisien a tidak boleh bernilai nol. Apabila a = 0, fungsi tersebut akan tereduksi menjadi fungsi linear.
- Koefisien a memiliki pengaruh paling signifikan, baik secara langsung maupun tunggal yang menentukan arah bukaan parabola dan jenis titik puncaknya.
- Jika nilai a positif (a>0), parabola akan terbuka ke atas dan memiliki titik balik minimum. Sebaliknya, jika nilai a negatif (a<0), parabola akan terbuka ke bawah dan memiliki titik balik maksimum. Dengan begitu, kita dapat menentukan gambaran visual tentang bentuk dasar kurva dengan lebih mudah.
- Koefisien tidak secara eksplisit muncul sebagai titik pada grafik, tetapi berperan penting dalam menentukan posisi sumbu simetri dan koordinat titik puncak melalui rumus-rumus turunan. Sedangkan koefisien c secara langsung menentukan titik potong kurva dengan sumbu-y. Hal ini dapat diketahui dengan menyubstitusikan nilai x=0 ke dalam fungsi, yang menghasilkan f(0)=a(0)2+b(0)+c=c. Oleh karena itu, titik potong sumbu-y selalu berada pada koordinat (0,c).

Diskriminan

```
>$&D=b^2-4*a*c
```

$$D = b^2 - 4\,a\,c$$

Titik Puncak

Titik puncak yang juga dikenal sebagai titik balik atau titik ekstrem adalah titik di mana parabola mencapai nilai maksimum atau minimumnya.

```
>$&xp=-b/(2*a)
```

$$xp = -\frac{b}{2\,a}$$

```
>$&yp=-D/(4*a)
```

$$yp = -\frac{D}{4\,a}$$

Contoh Penyelesaian

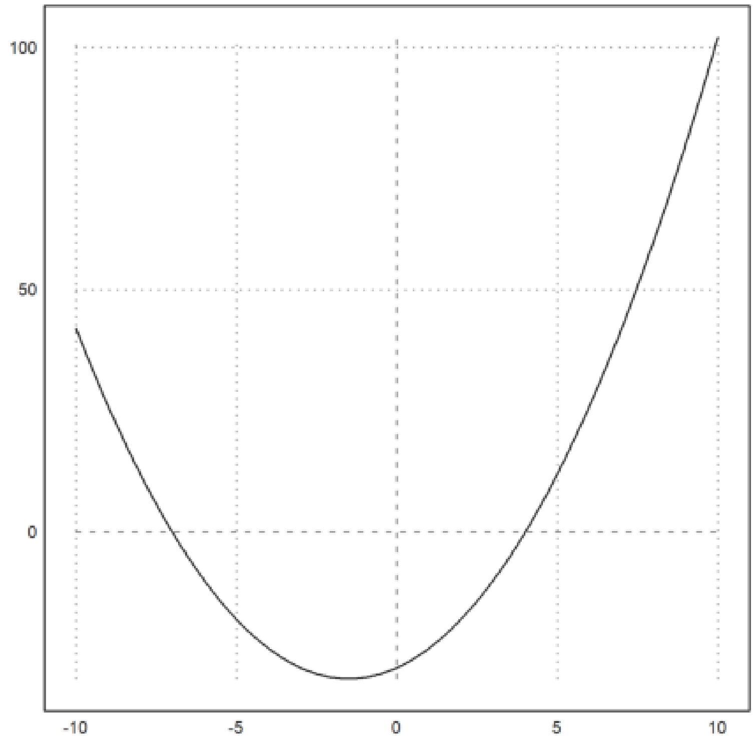
Tentukan penyelesaian dan grfaik dari soal berikut berikut.

$$x^2 + 3x - 28 = 0$$

```
>$&solve(x^2+3*x-28=0)
```

$$[x = 4, x = -7]$$

```
>plot2d("x^2+3*x-28", -10,10):
```



POLINOMIAL

Polinomial atau suku banyak adalah ekspresi matematika yang terdiri dari variabel dan koefisien yang digabungkan menggunakan operasi penjumlahan, pengurangan, dan perkalian. Variabel dalam polinomial hanya boleh memiliki pangkat berupa bilangan bulat non-negatif (0, 1, 2, 3, dst.). Singkatnya, polinomial merupakan bentuk aljabar yang memiliki dua atau lebih suku dengan pangkat bulat non negatif.

Bentuk Umum

```
>$a3*x^n+a2*x^(n-1)+a1*x^(n-2)+ax+a,
```

$$a_3\, x^n + a_2\, x^{n-1} + a_1\, x^{n-2} + ax + a$$

Derajat Polinomial

Derajat polinomial diambil dari pangkat tertinggi variabel dalam suatu ekspresi polinomial. Pangkat ini harus merupakan bilangan bulat non-negatif dan koefisien dari suku berpangkat tertinggi tersebut tidak boleh nol.

Derajat polinomial membantu mengklasifikasikan polinomial, seperti:
> Derajat 1: Polinomial linear (grafiknya garis lurus)
> Derajat 2: Polinomial kuadratik (grafiknya parabola)
> Derajat 3: Polinomial kubik
> Derajat 4: Polinomial kuartik

Jenis Polinomial Berdasarkan Jumlah Suku

1. Monomial (satu suku)

```
>$5*x
```

$$5\,x$$

2. Binomial (dua suku)

```
>$2*x+1
```

$$2\,x+1$$

3. Trinomial (tiga suku)

```
>$x^2+5*x+5
```

$$x^2+5\,x+5$$

Berikut ini adalah contoh non polinomial. Suku berikut termasuk non polinomial karena variabelnya berpangkat negatif.

```
>$x^(-3)
```

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

Penjumlahan, Pengurangan, Perkalian, dan Pembagian Polinomial

1. Tentukan penjumlahan polinomial berikut.

$$5x^3+3x^2+4x+1+7x^2+2x+5$$

```
>p=[5, 3, 4, 1]
```

```
         5.00         3.00         4.00         1.00
```

```
>q=[0, 7, 2, 5]
```

```
         0.00         7.00         2.00         5.00
```

```
>p+q
```

```
          5.00         10.00          6.00          6.00
```

Maka hasil penjumlahannya adalah sebagai berikut.

$$5x^3 + 10x^2 + 6x + 6$$

2. Tentukan hasil pengurangan polinomial berikut.

$$3x^2 - 7x - 2 - (5x^3 + 3x^2 + 4x - 6)$$

```
>$(3*x^2-7*x-2)-(5*x^3+3*x^2+4*x-6)
```

$$-5\,x^3 - 11\,x + 4$$

3. Tentukan hasil perkalian dari polinomial berikut.

$$(6x - 3)(2x + 5)$$

```
>t=[6, -3]
```

```
          6.00         -3.00
```

```
>u=[2, 5]
```

```
          2.00          5.00
```

```
>polymult(t, u)
```

```
         12.00         24.00        -15.00
```

Maka hasil perkalian tersebut adalah sebagai berikut.

$$12x^2 + 24x - 15$$

4. Tentukan hasil pembagian polinomial berikut.

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

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

```
        1.00        -1.00        -1.00        1.00
```

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

```
        1.00        -1.00
```

```
>polydiv(v,w)
```

```
        1.00         0.00        -1.00
```

Maka hasil pembagiannya adalah sebagai berikut.

$$x^2 - 1$$

EKSPONEN

Eksponen adalah konsep matematika yang digunakan untuk menyatakan perkalian berulang dari suatu bilangan atau variabel. Jika sebuah bilangan atau variabel n dikalikan dengan dirinya sendiri sebanyak n kali.

Sifat Eksponen

1. jika a^n = a^m dan a>0, a tidak sama dengan 1 maka m = n
2. jika a^n * a^m = 1, maka n+m = 0
3. Eksponen hanya berlaku jika variabel basis bernilai real dan tidak nol untuk pangkat negatif.

Penyelesaian Soal

1. Tentukan x yang memenuhi persamaan eksponen berikut.

$$3(x+1) + 3^x = 108$$

```
>$&solve(3^(x+1) + 3^x - 108=0, x)
```

$$\left[x = \frac{\log 27}{\log 3}\right]$$

```
>$float(solve(3^x=27, x))
```

$$[x = 3.0]$$

LOGARITMA

Logaritma merupakan invers atau kebalikan dari eksponen yang digunakan untuk menentukan besaran pangkat pada sebuah bilangan pokok. Secara umum bentuk logaritma terdiri dari tiga bagian yaitu basis, numerus, dan hasil logaritma.

Bentuk umumnya adalah sebagai berikut.

$$a^c = b$$

```
>$solve(a^c=b, c)
```

$$\left[c = \frac{\log b}{\log a}\right]$$

VEKTOR

Vektor merupakan besaran yang mempunyai arah. Secara geometri, setiap vektor dinyatakan secara geometris sebagai segmen garis berarah pada bidang atau ruang, dengan notasi garis berpanah. Ekor panah garis tersebut merupakan titik awal vektor, sedangkan ujung panah sebagai titik akhir (ujung) vektor tersebut.

Sifat Operasi Vektor

Jika u,v dan w adalah vektor-vektor di dalam ruang vektor (ruang 2 atau ruang 3) dan k dan l adalah skalar, maka hubungan yang berikut akan berlaku;
1. u+v = v+u
2. (u+v)+w = u+(v+w)
3. u+0= 0+u=0
4. u+ (-u)= 0
5. k*(l*u)= (k*l)*u
6. k(u+v) = k*u + k*v
7. (k+l)*u= k*u + l*u

Panjang Vektor

Panjang Vektor dapat dicari dengan menggunakan fungsi norm[v] pada EMT.
Contoh:
Tentukan panjang vektor dari vektor berikut.

$$(-3, 2, 1)$$

```
>$& u:= sqrt (u1^2 + u2^2 +u3^2)
```

$$\sqrt{{u_{3}}^2+{u_{2}}^2+{u_{1}}^2}$$

```
>$ u= sqrt((-3*-3)+(2^2)+(1^2))
```

$$\sqrt{{u_{3}}^2+{u_{2}}^2+{u_{1}}^2}=\sqrt{14}$$

Hasil Kali Titik

Hasil kali titik merupakan operasi antara dua buah vektor yang akan menghasilkan skalar.Pada software EMT menggunakan perintah dot(u,v) atau u*v

Contoh:

Tentukan hasil kali titik antara dua vektor berikut.

$$A = 2i + 3j + 5k$$

$$B = 4i + 2j - k$$

```
>$&A:= [2;3;5]
```

$$\begin{pmatrix} 2 \\ 3 \\ 5 \end{pmatrix}$$

```
>$&B:= [4;2;-1]
```

$$\begin{pmatrix} 4 \\ 2 \\ -1 \end{pmatrix}$$

```
>$& (A*B)
```

$$\begin{pmatrix} 8 \\ 6 \\ -5 \end{pmatrix}$$

PENYELESAIAN SOAL ALJABAR

1. Tentukan penyelesaian dari soal berikut.

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

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

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

Penyelesaian atau penyederhanaan soal di atas dilakukan dengan mengoperasikan suku dengan variabel yang sama.
- Untuk variabel a, sesuai dengan aturan pembagian eksponen, maka perhitungan dilakukan dengan mengurangkan pangkatnya, maka:

$$\frac{a^{10}}{a^{6}}$$

Didapatkan hasilnya sebagai berikut

$$a^{4}$$

- Untuk variabel b, sesuai dengan aturan pembagian eksponen, maka perhitungan dilakukan dengan mengurangkan pangkatnya, maka:

$$\frac{b^{-10}}{b^{-3}}$$

Didapatkan hasilnya sebagai berikut

$$b^{-5}$$

- Untuk variabel c, sesuai dengan aturan pembagian eksponen, maka perhitungan dilakukan dengan mengurangkan pangkatnya, maka:

$$\frac{c^{7}}{c^{5}}$$

Didapatkan hasilnya sebagai berikut

$$c^{2}$$

- Untuk konstanta, maka:

$$\frac{24}{12}$$

Didapatkan hasilnya adalah 2
Setelah itu, hasilnya masih perlu dipangkatkan sebagai berikut.

$$\left(2a^4b^{-5}c^2\right)^{-5}$$

Didapatkan hasilnya yaitu:

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

2. Hitunglah operasi di bawah ini!

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

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

$$-5$$

Untuk menyelesaikan soal di atas, maka kita dahulukan operasi yang ada di dalam kurung, lalu dilanjutkan dengan perpangkatan, perkalian dan pembagian, lalu penjumlahan dan pengurangan.
3. James menyetorkan \$250 ke rekening pensiun setiap bulan mulai usia 40 tahun. Jika investasi tersebut menghasilkan bunga 5% dengan penggandaan bulanan, berapa banyak yang akan terkumpul di rekening itu ketika ia pensiun 27 tahun kemudian?

```
>p:= 250
```

```
       250.00
```

```
>b:= 0.05/12
```

```
         0.00
```

```
>n:= 27*12
```

```
       324.00
```

```
>$&p * (((1+b)^n - 1) / b) * (1+b)
```

$$\frac{(b+1)\,\left((b+1)^n-1\right)\,p}{b}$$

```
>f:=p * (((1+b)^n - 1) / b) * (1+b)
```

```
171508.96
```

Soal di atas merupakan soal bunga bank. Untuk menyelesaikan soal dengan rumus bunga bank yang disetorkan di awal bulan, maka dilakukan dengan perhitungan berikut.

$$FV = P \cdot \frac{(1+b)^n - 1}{b} \cdot (1+b)$$

Sebelumnya kita perlu mendefinisikan terlebih dahulu p,b, dan n beradasarkan informasi pada soal, dimana:
- p = setoran bulanan
- b = bunga bulanan
- n = lama waktu (bulan)

4. Selesaikan soal berikut.

$$(3x^2 - 2x - x^3 + 2) - (5x^2 - 8x - x^3 + 4)$$

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

```
-1.00          3.00          -2.00          2.00
```

```
>B = [-1,5,-8,4]
```

```
-1.00          5.00          -8.00          4.00
```

```
>A-B
```

```
0.00          -2.00          6.00          -2.00
```

Soal tersebut dapat diselesaikan dengan mengubahnya menjadi sebuah vektor koefisien. Dengan begitu kita perlu mendefinisikan A dan B, dimulai dari dari koefisien dengan pangkat tertinggi. Lalu hasilnya dapat kita ubah kembali menjadi bentuk polinomial, sehingga hasilnya adalah

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

5. Faktorkan persamaan di bawah ini.

$$x^2 + 12x + 36 = 0$$

```
>$&factor(x^2+12*x+36=0)
```

$$(x+6)^2 = 0$$

Dengan melalui EMT, fungsi factor() digunakan untuk menentukan factorisasi dari sebuah persamaan. Dari perhitungan yang telah dilakukan didapatkan

$$(x+6)^2$$

Maka artinya faktorisasi persamaan tersebut adalah

$$(x+6)(x+6)$$

6. Selesaikan persamaan berikut.

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

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

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

Untuk dapat menemukan nilai x yang memenuhi persamaan, maka digunakan fungsi solve("persamaan",variabel). Untuk mendapatkan hasil dengan tampilan lebih baik, maka dapat ditambahkan $ pada bagian awal perintah. Dengn begitu didapatkan hasil bahwa x yang memenuhi adalah -3/2.

7. Selesaikan operasi di bawah ini.

$$\frac{6}{x+3} - \frac{x+4}{9-x^2} + \frac{2x-3}{9-6x+x^2}$$

```
>$&(6/(x+3) - (x+4)/(9-x^2) + (2*x-3)/(9-6*x+x^2))
```

$$\frac{2\,x-3}{x^2-6\,x+9} - \frac{x+4}{9-x^2} + \frac{6}{x+3}$$

```
>$&ratsimp(%)
```

$$x^2 + 12\,x + 36 = 0$$

Untuk menyelesaikan operasi variabel yang kompleks, fungsi yang digunakan yaitu:
- $&() = Untuk memerintahkan EMT menghitung atau menyederhanakan ekspresi secara simbolik
- % = Berarti hasil dari perintah sebelumnya
- ratsimp (expr) atau rational simplify = Untuk menyederhanakan pecahan rasional menjadi bentuk paling sederhana

8. Selesaikan operasi di bawah ini.

$$\frac{\frac{1}{x^2} - \frac{1}{y^2}}{\frac{1}{x^2} - \frac{2}{xy} + \frac{1}{y^2}}$$

```
>$&((1/x^2 - 1/y^2) / (1/x^2 - 2/(x*y) + 1/y^2))
```

$$\frac{\frac{1}{x^2}-\frac{1}{y^2}}{-\frac{2}{x\,y}+\frac{1}{y^2}+\frac{1}{x^2}}$$

```
>$&ratsimp(%)
```

$$\frac{y+x}{y-x}$$

Dengan cara yang sama dengan nomor 7, maka operasi ini dapat EMT selesaikan dengan fungsi ratsimp(%). Sehingga didapatkan hasilnya yaitu:

$$\frac{y-x}{x+y}$$

9. Faktorkan persamaan berikut.

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

```
>$&factor(12*z^2+z=6)
```

$$z\,\left(12\,z+1\right)=6$$

10. Selesaikan operasi berikut.

$$\frac{x^2-9}{x^3+27}\cdot\frac{5x^2-15x+45}{x^2-2x-3}+\frac{x^2+x}{4+2x}$$

```
>$&((x^2 - 9)/(x^3 + 27) * (5*x^2 - 15*x + 45)/(x^2 - 2*x - 3) + (x^2 + x)/(4 + 2*x))
```

$$\frac{\left(x^2-9\right)\,\left(5\,x^2-15\,x+45\right)}{\left(x^2-2\,x-3\right)\,\left(x^3+27\right)}+\frac{x^2+x}{2\,x+4}$$

```
>$&ratsimp(%)
```

$$\frac{x^3+2\,x^2+11\,x+20}{2\,x^2+6\,x+4}$$

11. Selesaikan operasi berikut.

$$\left[\frac{\frac{x+1}{x-1}+1}{\frac{x+1}{x-1}-1}\right]^5$$

```
>$&(((x+1)/(x-1)+1)/((x+1)/(x-1)-1))^5
```

$$\frac{\left(\frac{x+1}{x-1}+1\right)^5}{\left(\frac{x+1}{x-1}-1\right)^5}$$

```
>$&ratsimp(%)
```

$$x^5$$

12. Tentukan faktorisasi persamaan berikut.

$$t^2+12t+27=0$$

```
>$&factor(t^2+12*t+27=0)
```

$$(t+3)\ (t+9)=0$$

```
>$&solve(t^2+12*t+27=0, t)
```

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

13. Selesaikan persamaan berikut.

$$3(2n-5)-7=4(n-9)$$

```
>$&solve(3*(2*n-5)-7=4*(n-9), n)
```

$$[n=-7]$$

Dengan fungsi yang sama pula, solve() dapat membantu untuk menemukan nilai n yang memenuhi persamaan yang diberikan.

14. Faktorkan binomial berikut.

$$4p^2 + 8pq + 4q^2$$

```
>$&factor(4*p^2+8*p*q+4*q^2)
```

$$4\,\left(q+p\right)^2$$

15. Faktorkan soal berikut.

$$a^3b - 9a^2b^2 + 20ab^3$$

```
>$&factor(a^3*b-9*a^2*b^2+20*a*b^3)
```

$$a\,b\,\left(4\,b-a\right)\,\left(5\,b-a\right)$$

16. Selesaikan permasalahan berikut.

$$(-5m^4n^2)(6m^2n^3)$$

```
>$&(-5*m^4*n^2)*(6*m^2*n^3)
```

$$-30\,m^6\,n^5$$

Permasalahan di atas dapat dilakukan dengan melakukan perkalian sejeni, antara lain:
- Mengalikan koefisien, yaitu mengalikan -5 dengan 6, sehingga hasilnya adalah -30
- mengalikan pangkat sejenis m, yaitu:

$$(m^4)(m^2) = m^6$$

- Mengalikan pangkat sejenis n, yaitu:

$$(n^2)(n^3) = n^5$$

Sehingga hasil perkalian persoalan tersebut yaitu:

$$-30m^6n^5$$

17. Selesaikan permasalahan berikut.

$$(a-b)(2a^3-ab+3b^2)$$

```
>$&expand((a-b)*(2*a^3 - a*b + 3*b^2))
```

$$-3\,b^3+4\,a\,b^2-2\,a^3\,b-a^2\,b+2\,a^4$$

Dalam menyelesaikan soal ini diperlukan fungsi expand(). Expand dipakai untuk menguraikan ekspresi matematika yang masih dalam bentuk perkalian atau pangkat menjadi bentuk penjumlahan/polinom yang lengkap.

18. Selesaikan soal berikut.

$$(x^{3m}-t^{5n})^2$$

```
>$&expand((x^(3*m) - t^(5*n))^2)
```

$$x^{6\,m}-2\,t^{5\,n}\,x^{3\,m}+t^{10\,n}$$

19. Tentukan hasil yang paling sederhana.

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

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

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

Penyelesaian atau penyederhanaan soal di atas dilakukan dengan mengoperasikan suku dengan variabel yang sama.

- Untuk variabel p, sesuai dengan aturan pembagian eksponen, maka perhitungan dilakukan dengan mengurangkan pangkatnya, maka:

$$\frac{p^{12}}{p^8}$$

Didapatkan hasilnya sebagai berikut

$$p^4$$

- Untuk variabel q, sesuai dengan aturan pembagian eksponen, maka perhitungan dilakukan dengan mengurangkan pangkatnya, maka:

$$\frac{q^{-14}}{q^6}$$

Didapatkan hasilnya sebagai berikut

$$q^{-20}$$

- Untuk variabel r, sesuai dengan aturan pembagian eksponen, maka perhitungan dilakukan dengan mengurangkan pangkatnya, maka:

$$\frac{r^{22}}{r^-15}$$

Didapatkan hasilnya sebagai berikut

$$r^{37}$$

- Untuk konstanta, maka:

$$\frac{125}{25}$$

Didapatkan hasilnya adalah 2

Setelah itu, hasilnya masih perlu dipangkatkan sebagai berikut.

$$\left(5p^4q^{-20}r^{37}\right)^{-4}$$

Didapatkan hasilnya yaitu:

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

20. Tentukan himpunan penyelesaian dari pertidaksamaan berikut.

$$2y-3 \geq 1-y+5$$

```
>&load(fourier_elim)
```

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

```
>$&fourier_elim([2*y - 3 >= 1 - y + 5], [y])
```

$$[y = 3] \vee [3 < y]$$

Untuk menyelesaikan soal nomor 20 di EMT, maka harus memuat paket fourier_elim terlebih dahulu. Fourier_elim digunakan untuk menyelesaikan sistem pertidaksamaan linear, khususnya untuk menghilangkan satu atau lebih variabel dari sistem tersebut.

21. Selesaikan soal berikut.

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

```
>$&(2^6 * 2^(-3) / 2^10 / 2^(-8))
```

$$2$$

Dalam EMT, $ (program maxima) digunakan untuk mengevaluasi ekspresi. Sedangkan & adalah operator yang sering digunakan untuk memanggil fungsi secara tidak langsung. Setelah dioperasikan maka jawaban soal ini adalah 2.

22. Sederhanakan soal berikut.

$$\frac{(3x^a y^b)^3}{(-3x^a y^b)^2}$$

```
>$&((3*x^a*y^b)^3 / (-3*x^a*y^b)^2)
```

$$3\,x^a\,y^b$$

23. Sederhanakan bentuk berikut.

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

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

$$(x-1)\,(x+1)\,\left(x^2+1\right)$$

```
>$&ratsimp(%)
```

$$x^4 - 1$$

Dalam soal ini, fungsi yang digunakan adalah ratsimp(%). ratsimp (%) digunakan untuk menyederhanakan ekspresi aljabar. Di mana % merujuk pada hasil dari perintah sebelumnya.

24. Selesaikan soal berikut.

$$(5x^2 + 4xy - 3y^2 + 2) - (9x^2 - 4xy + 2y^2 - 1)$$

```
>$&((5*x^2 + 4*x*y - 3*y^2 + 2) - (9*x^2 - 4*x*y + 2*y^2 - 1))
```

$$-5\,y^2 + 8\,x\,y - 4\,x^2 + 3$$

25. Faktorkan bentuk kuadrat berikut.

$$4ax^2 + 20ax - 56a$$

```
>$&factor(4*a*x^2 + 20*a*x - 56*a)
```

$$4\,a\,\left(x-2\right)\,\left(x+7\right)$$

Unuk dapat memfaktorkan bentuk kuadrat di EMT, maka dibutuhkan fungsi factor(), tujuannya adalah mengubahnya dari bentuk perkalian yang diperluas menjadi bentuk perkalian dari faktor-faktornya. Fungsi factor() akan mencari faktor persekutuan terbesar (FPB) dari semua suku dan kemudian memfaktorkan polinomial yang tersisa.

26. Factorkan bentuk berikut.

$$1 - 8x + 16x^2$$

```
>$&factor(1 - 8*x + 16*x^2)
```

$$\left(4\,x-1\right)^2$$

27. Diberikan

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

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

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

Tentukan

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

```
>function f(x)&=3*x+1
```

```
                          3 x + 1
```

```
>function g(x)&=x^2-2*x-6
```

```
                           2
                          x  - 2 x - 6
```

```
>$&f(g(-1))
```

$$-8$$

Untuk menyelesaikan soal tersebut, kita perlu definisikan dulu fungsi yang diberikan untuk selanjutnya dapat dicari nilainya yang ditanyakan.

28. Selesaikan soal berikut. Tulis jawabannya dalam bentuk a+bi

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

```
>$&(-5+3*i) + (7+8*i)
```

$$11\,i+2$$

29. Tentukan solusi yang memenuhi.

$$x^2-2x=15$$

```
>$&solve(x^2-2*x=15, x)
```

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

Dengan menggunakan EMT kita dapat gunakan fungsi solve() untuk menemukan solusi atau nilai x yang memenuhi persamaan. Dari perhitungan didapatkan nilai x yang memenuhi adalah x=-3 dan x=5

30. Sketsakan grafik fungsi polinomial berikut.

$$f(x) = x^3 - 7x + 6$$

```
>plot2d("x^3-7*x+6"):
```