

Наръчник на учителя Системи за компютърна алгебра



Петер Кортеси
Университет на Мишколц, Унгария

Запознаване със системите за компютърна алгебра (СКА)

Названието, използвано за средствата за компютърна алгебра е Системи за Компютърна Алгебра (СКА), тъй като освен възможностите за числени и символични изчисления, те могат да се използват за редактиране на математически текст, за визуализиране на математически модели, за създаване на програми и др.

Най-известните СКА са *MAPLE* и *MATHEMATICA*.

По-надолу ще представим примери с *MAPLE 9*, макар че с бързото развитие на математическия софтуер най-новите версии са *MAPLE 11* и *Mathematica 6*.

Кратък преглед на някои други системи за компютърна алгебра

Освен по-горе споменатите системи съществуват много други програмни средства за компютърна алгебра – общи и специализирани.

От общ характер са например: MUPAD, DERIVE, MATLAB, MATHCAD, които работят повече или по-малко по еднаква концепция, всички могат да извършват числени и символни изчисления, като всяка от тях предлага някои преимущества.

Специализирани СКА са: ODE, DELiA – за решаване на обикновени диференциални уравнения, SPSS – за статистика в икономическите и социални науки, CAYLEY, LiE и GAP – за алгебра, CAMAL, SCHONSCHIP и STENSOR за физика, а също и такива като MACSYMA, REDUCE и др.

Нека започнем с MAPLE

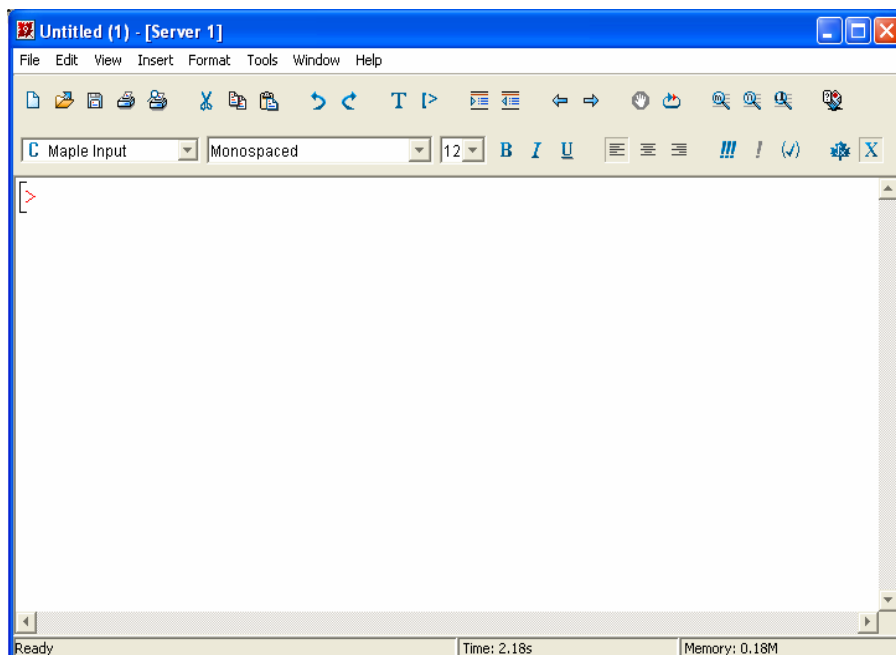
Ще започнем с програмата MAPLE, с представяне на нейното лого - листо:



MAPLE дава възможност за отваряне на няколко работни листа, първият от които се отваря автоматично със стартиране на програмата. Те се запомнят във формат FILENAME.ms. За простота ние ще запомняме нашите листи като файл, наречен ECADL-nr.ms.

Работно пространство на Maple

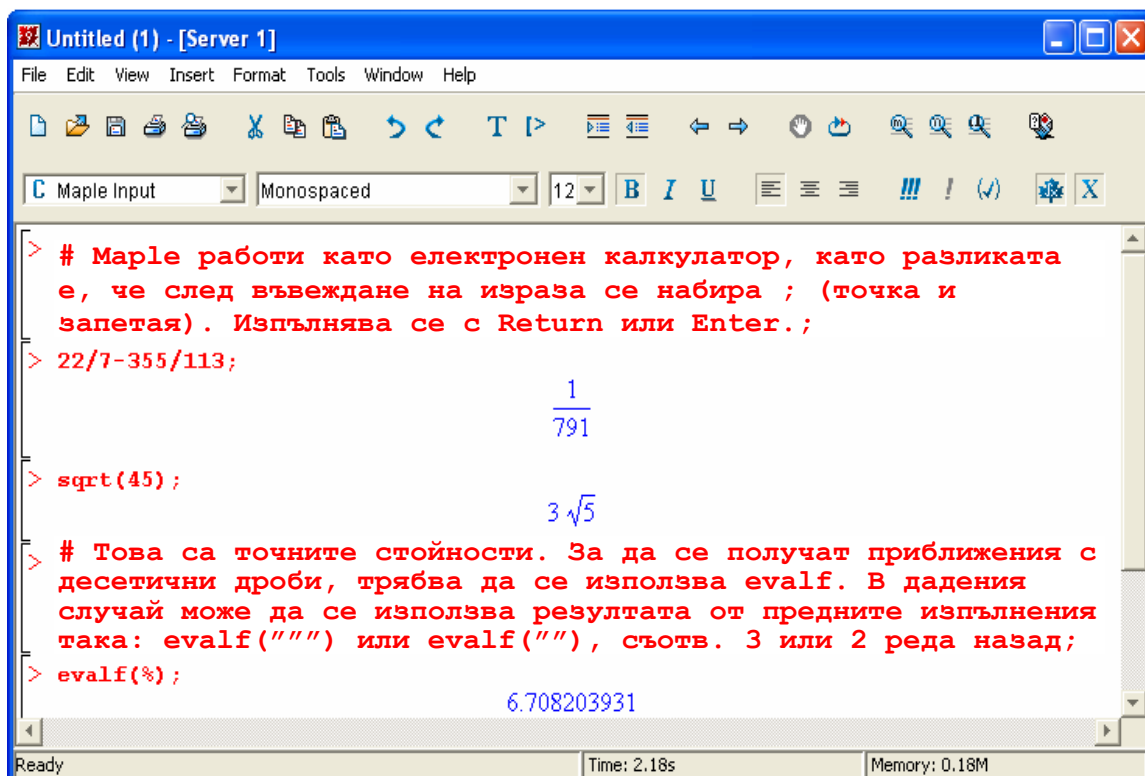
Съдържа общия прозорец, отварящ по подразбиране работен лист с името Untitled (1). Повечето от редовете на основните менюта са подобни на тези на Word. Има и такива за математика, но най-много използваното меню е може би менюто Help (Система за Помощ). Може да отворите едновременно няколко работни листа, те ще се именуват Untitled (2) и т.н., но само един от тях ще е активен (върху който сте кликнули).



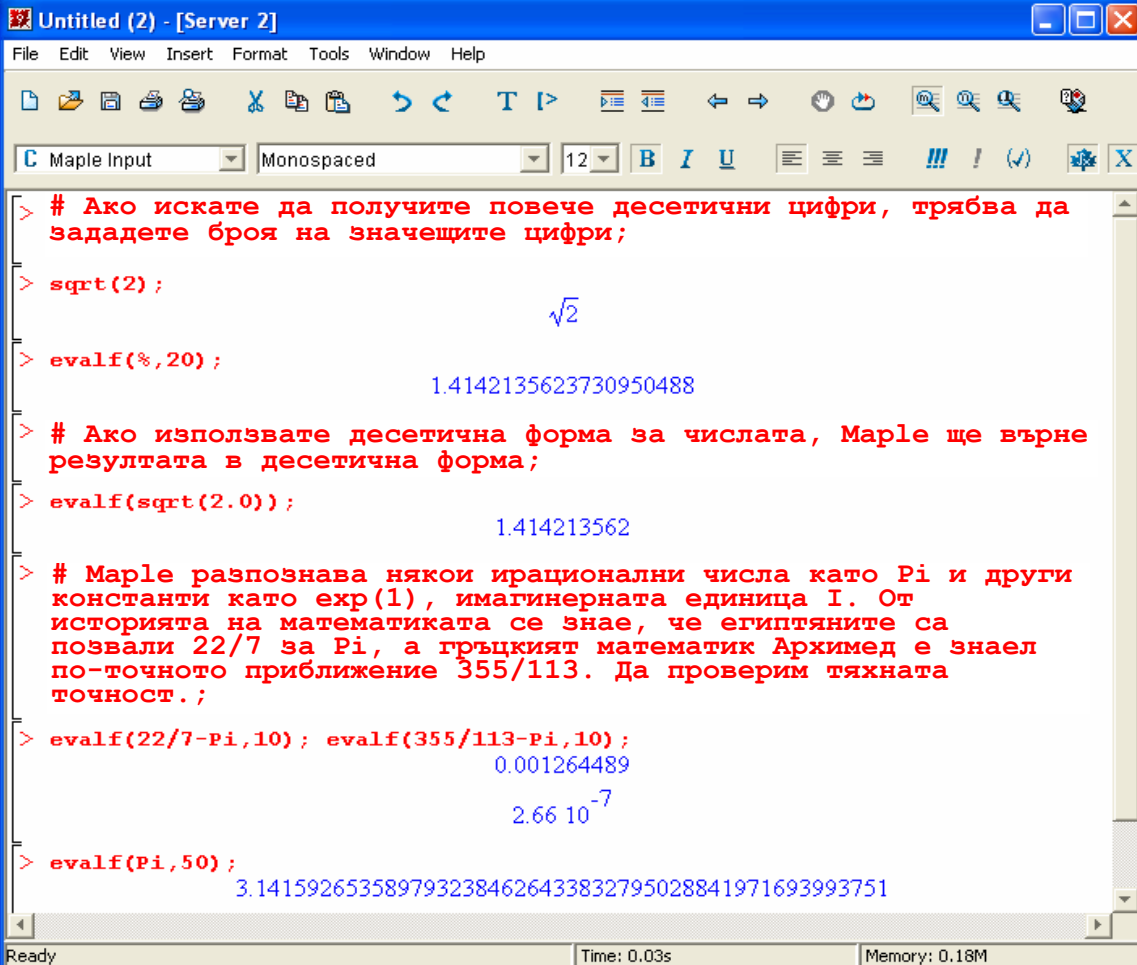
Стартиране на MAPLE

Всеки ред се стартира автоматично с подканващия символ `>`, след който можете да набирате своя текст. В прозорец на Maple можете да въвеждате нематематически текст, коментари, като напишете `#` текст и завършите с точка и запетая ;

Независимо от дължината на въведения команден ред, за да го потвърдите, натиснете Return или Enter.

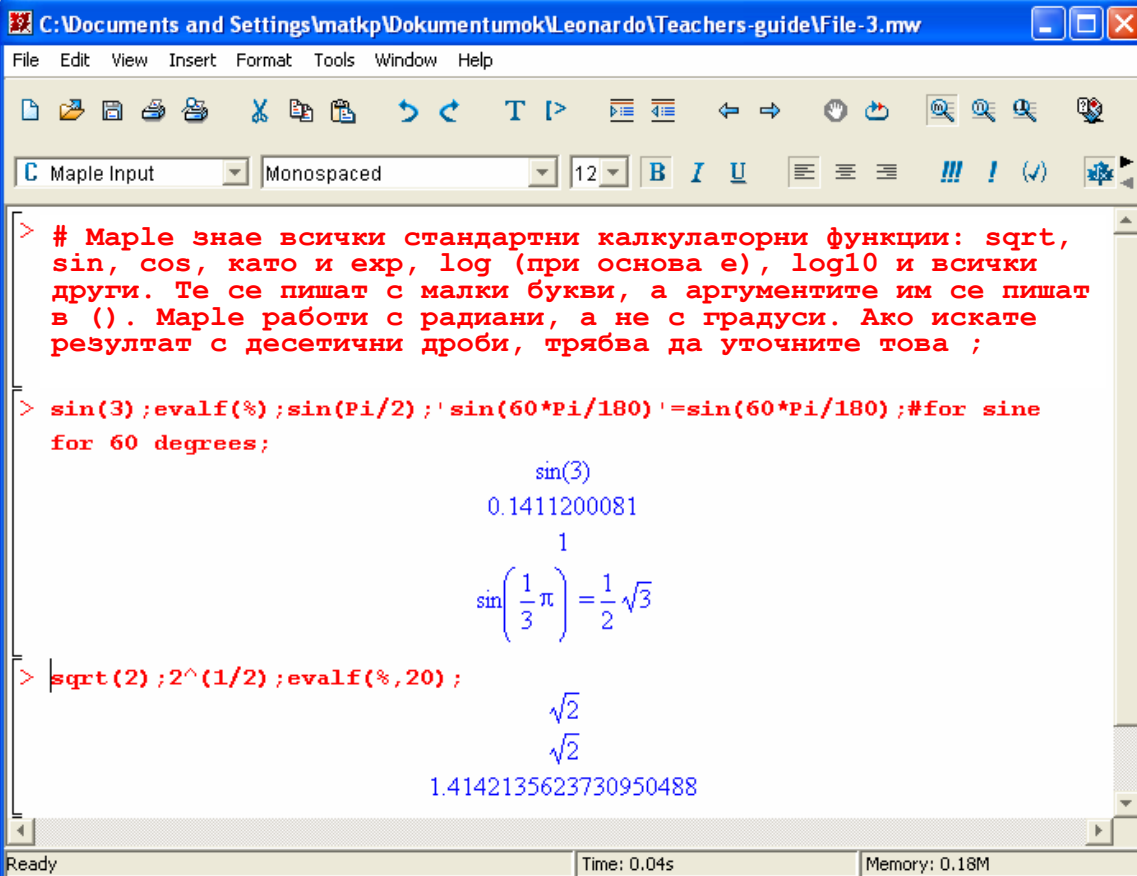


Повече десетични знаци и функции в MAPLE



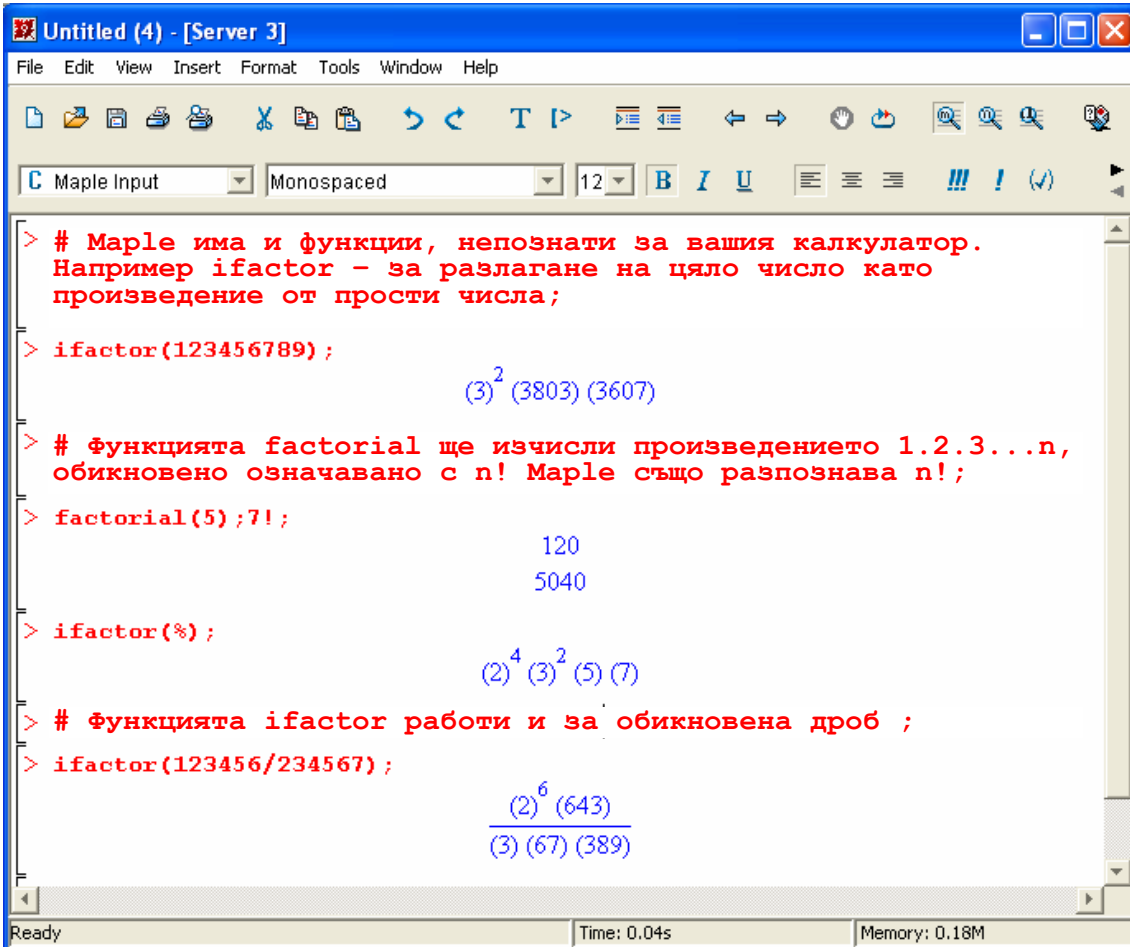
```
Untitled (2) - [Server 2]
File Edit View Insert Format Tools Window Help
Maple Input Monospaced 12 B I U
> # Ако искате да получите повече десетични цифри, трябва да
    зададете броя на значещите цифри;
> sqrt(2);
       $\sqrt{2}$ 
> evalf(%,20);
      1.4142135623730950488
> # Ако използвате десетична форма за числата, Maple ще върне
    резултата в десетична форма;
> evalf(sqrt(2.0));
      1.414213562
> # Maple разпознава някои ирационални числа като Pi и други
    константи като exp(1), имагинерната единица I. От
    историята на математиката се знае, че египтяните са
    позволили 22/7 за Pi, а гръцкият математик Архимед е знаел
    по-точното приближение 355/113. Да проверим тяхната
    точност.;
> evalf(22/7-Pi,10); evalf(355/113-Pi,10);
      0.001264489
      2.66 10-7
> evalf(Pi,50);
      3.1415926535897932384626433832795028841971693993751
Ready Time: 0.03s Memory: 0.18M
```

Стандартни изчислителни функции в Maple



```
C:\Documents and Settings\matkp\Работен стол\Leonardo\Teachers-guide\file-3.mw
File Edit View Insert Format Tools Window Help
Maple Input Monospaced 12 B I U
> # Maple знае всички стандартни калкулаторни функции: sqrt,
    sin, cos, като и exp, log (при основа e), log10 и всички
    други. Те се пишат с малки букви, а аргументите им се пишат
    в (). Maple работи с радиани, а не с градуси. Ако искате
    резултат с десетични дробни, трябва да уточните това ;
> sin(3);evalf(%) ;sin(Pi/2); 'sin(60*Pi/180) '=sin(60*Pi/180);#for sine
    for 60 degrees;
      sin(3)
      0.1411200081
      1
       $\sin\left(\frac{1}{3}\pi\right) = \frac{1}{2}\sqrt{3}$ 
> sqrt(2);2^(1/2);evalf(%,20);
       $\sqrt{2}$ 
       $\sqrt{2}$ 
      1.4142135623730950488
Ready Time: 0.04s Memory: 0.18M
```

Други функции



```
Untitled (4) - [Server 3]
File Edit View Insert Format Tools Window Help
C Maple Input Monospaced 12 B I U
> # Maple има и функции, непознати за вашия калкулатор.
  Например ifactor - за разлагане на цяло число като
  произведение от прости числа;
> ifactor(123456789);
      (3)2 (3803) (3607)
> # Функцията factorial ще изчисли произведението 1.2.3...n,
  обикновено означавано с n! Maple също разпознава n!;
> factorial(5); 7!;
      120
      5040
> ifactor(%);
      (2)4 (3)2 (5) (7)
> # Функцията ifactor работи и за обикновена дроб ;
> ifactor(123456/234567);
      (2)6 (643)
      (3) (67) (389)
Ready Time: 0.04s Memory: 0.18M
```

Система за помощ в Maple

За да видите файловете за помощ за някоя команда на Maple, напишете командата и я маркирайте. Или запишете въпросителен знак ? и командата. Няма нужда от ; (точка и запетая).

Можете да изберете и директно команда от Help. В края на файла ще намерите някои примери по нейното използване. Тези редове могат да се копират и вмъкват в работния лист със стандартните команди Copy, Paste. Впоследствие можете да ги промените според вашите нужди. Всеки помощен файл предлага и връзки към близки термини в най-отдолу, които да ви помогнат да намерите точно каквото искате.

Менюто Help има също и възможност за Пълно търсене на текст ("Full text search"), за да издирите всички думи и фрази, които имате предвид.

Да използваме менюто Help за да видим какво е описанието за функцията evalf

В менюто Help има основно дърво, както и самия help файл като хипертекст, за да може бързо да се търси вътре в помощното меню.

Maple 9 Help - [evalf]

File Edit View Help

Contents Topic Search History

- Discrete Mathematics
- Evaluation
 - allvalues
 - assume
 - cost
 - eval
 - Eval
 - evala
 - evalb
 - evalc
 - evalf**
 - evalhf
 - evalindets
 - evalm
 - evaln
 - evalpow
 - evalr
 - evalrC
 - hfarray
 - last name evaluation
 - special evaluation
 - value
- Factorization and Solving Equations
- Financial Functions
- FunctionAdvisor
- General Information
- Geometry

evalf - evaluate using floating-point arithmetic

Calling Sequence

```
evalf(expr)
evalf[n](expr)
```

Parameters

expr - any expression

n - (optional) integer; specifies number of digits

Description

- The **evalf** function evaluates to floating-point or complex floating-point numbers, expressions (or subexpressions) involving constants such as Pi, exp(1), gamma, and functions such as exp, ln, sin, arctan, cosh, GAMMA, and erf. For the complete list of known constants, see [minames](#). For the complete list of known functions, see [nifcns](#).
- The accuracy of the result is determined by the value of the environment variable **Digits**. By default the results will be computed using 10-digit floating-point arithmetic, since the initial value of **Digits** is 10. A user can change the value of **Digits** to any positive integer that does not exceed the value given by `kernelopts(maxdigits)`.

evaluate using floating-point arithmetic

Maple 9 Help - [evalf]

File Edit View Help

Contents Topic Search History

- Discrete Mathematics
- Evaluation
 - allvalues
 - assume
 - cost
 - eval
 - Eval
 - evala
 - evalb
 - evalc
 - evalf**
 - evalhf
 - evalindets
 - evalm
 - evaln
 - evalpow
 - evalr
 - evalrC
 - hfarray
 - last name evaluation
 - special evaluation
 - value
- Factorization and Solving Equations
- Financial Functions
- FunctionAdvisor
- General Information
- Geometry
- Group Theory
- Inert Functions

Examples

```
> evalf(Pi);
3.141592654

> evalf[15](5/3*exp(-2+3*I)*sin(Pi/4));
-0.157898022493763 + 0.0225078172647505 I

> evalf(cos(1) + sin(1)*I);
0.5403023059 + 0.8414709848 I

> evalf(3/4*x^2+1/3*x-sqrt(2));
0.7500000000 x^2 + 0.3333333333 x - 1.414213562

> sin(3.5+4.5*I);
-15.79019836 - 42.14337074 I

> int(exp(x^3), x=0..1);
∫₀¹ e^(x³) dx

> evalf(%);
```

evaluate using floating-point arithmetic

Двумерна графика - 2D

Има няколко възможности за използване структурата на функцията plot в Maple - 2D, 3D и анимирани графики.

За 2D графика се задава командата:

plot (f, h, v, options);

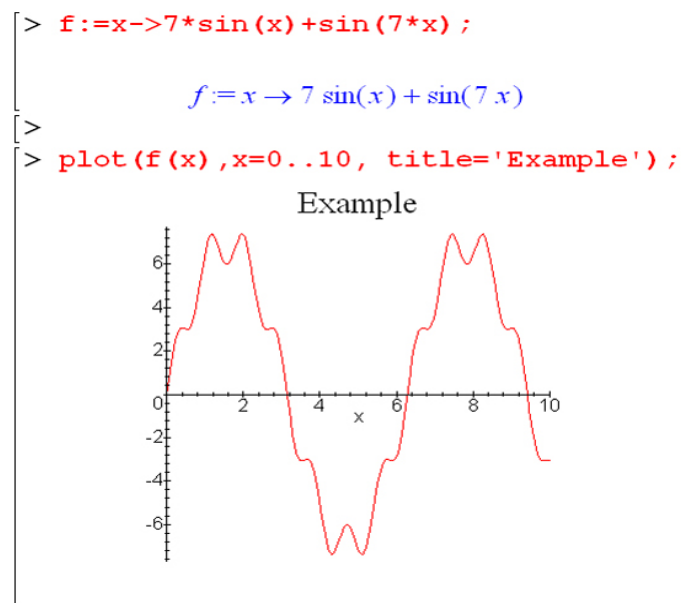
f функцията, която да се нарисува

h хоризонтална наредба

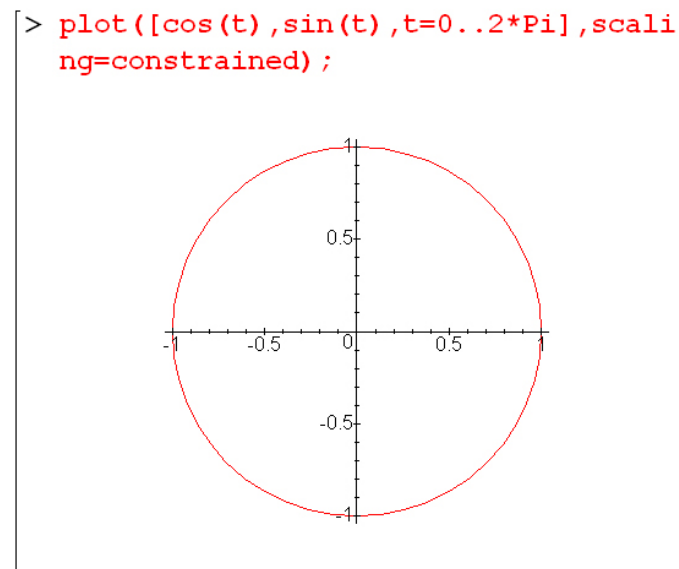
v вертикална наредба (възможност)

възможностите са: axes, color, labels, scaling, title, style, symbol, и др.

Рисуване на 2D- общ пример: дефинира се функция и се нарисува графиката ѝ с командата plot:



Рисуване на 2D графика – Пример с параметризирана функция



Рисуване на 2D графика – Полярни координати

Декартовите координати са стандартни за представяне на точкова графика в Maple. Друга възможност са полярните координати (r, Θ) . Трябва да се използва специален пакет `with(plots)`.

Командата е: **`polarplot (r, Θ , options);`**

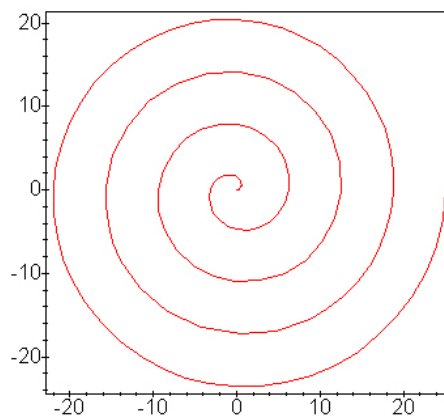
`r` дължина на радиус вектора на точката
 `Θ` най-близкият ъгъл до оста `Ox`

Рисуване на 2D графика – Пакет `with(plots)`:

```
> with (plots);  
[animate, animate3d, changecoords, complexplot,  
complexplot3d, conformal, contourplot,  
contourplot3d, coordplot, coordplot3d, cylinderplot,  
densityplot, display, display3d, fieldplot, fieldplot3d,  
gradplot, gradplot3d, implicitplot, implicitplot3d,  
inequal, listcontplot, listcontplot3d, listdensityplot,  
listplot, listplot3d, loglogplot, logplot, matrixplot,  
odeplot, pareto, pointplot, pointplot3d, polarplot,  
polygonplot, polygonplot3d, polyhedraplot, replot,  
rootlocus, semilogplot, setoptions, setoptions3d,  
spacecurve, sparsematrixplot, sphereplot, surfdata,  
textplot, textplot3d, tubeplot]
```

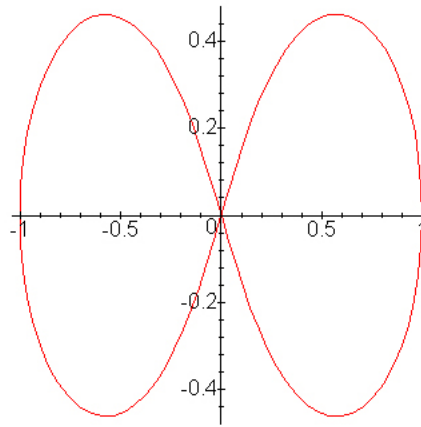
Рисуване на 2D графика – Полярни координати: Пример 1

```
> polarplot (theta, theta=0..8*Pi, scalin  
g=constrained, axes=boxed);
```



Рисуване на 2D графика – Полярни координати: Пример 2

```
> polarplot([sin(t),cos(t),t=0..2*Pi];
```



Рисуване на 2D графика – Прекъснати функции

Внимавайте с прекъснатите функции. Следната функция $f(x)$ има две точки на прекъсване: $x=1$ и $x=2$:

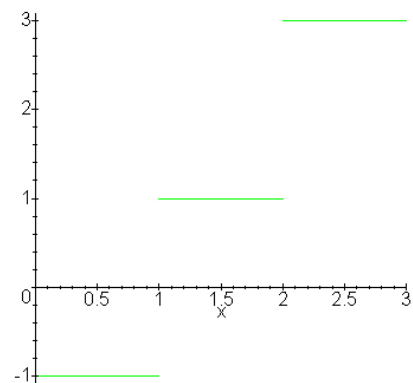
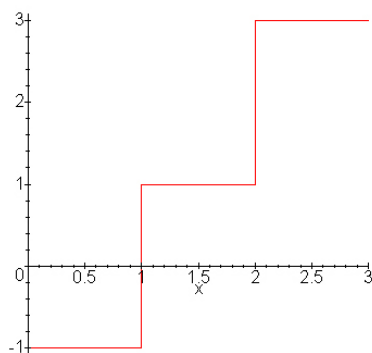
$$f(x) = \begin{cases} -1 & x \leq 1 \\ 1 & 1 \leq x \leq 2 \\ 3 & 2 \leq x \end{cases}$$

Maple ще покаже перпендикулярни прави в точките на прекъсване.

С използване на опцията `discont=true` maple ще разпознае прекъснатата функция.

Рисуване на 2D графика – Прекъснатата функция:

```
> f:=x->piecewise(x<1, -1, x<2, 1,3); > plot(f(x),x=0..3,discont=true);  
f:=x -> piecewise(x < 1, -1, x < 2, 1, 3)  
> plot(f(x),x=0..3);
```



Рисуване на 2D графика – Няколко функции на една и съща графика

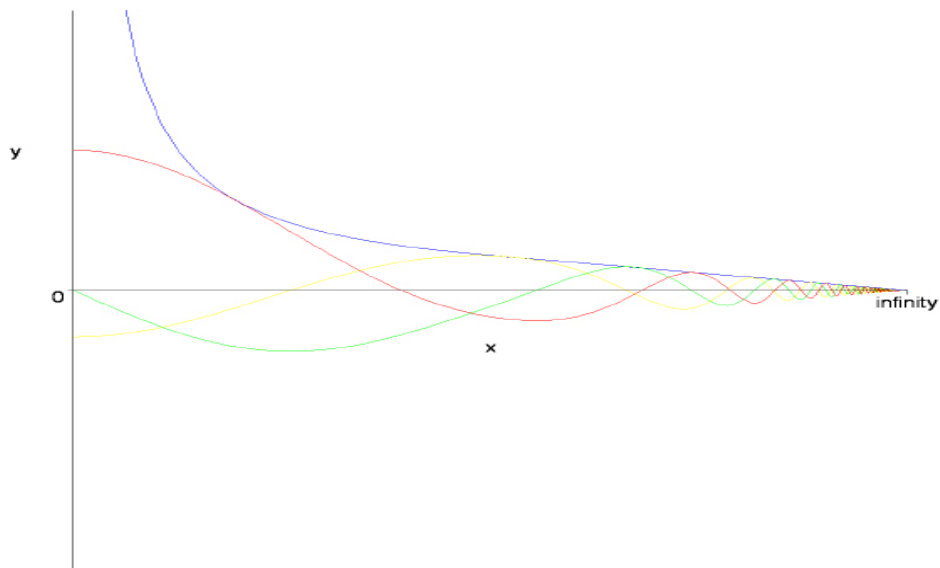
Командата е: `plot ([function1, function2, ...], parameter, options);`

Ако пишете `infinity` няма нужда да се задава оста. Смяната на цвета е автоматична.

Може да се промени с опции. Също така може да се промени и стилът на линиите.

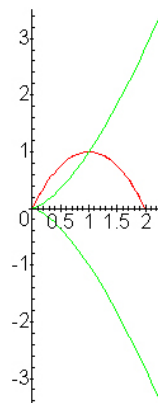
Рисуване на 2D графика – Няколко функции: Пример 1

```
> plot ([f(x), diff(f(x), x$2), 1/x], x=0..infinity, y=-2..2);
```



Рисуване на 2D графика – Няколко функции: Пример 2

```
> plot ([[1-cos(t), sin(t)^(2), t=0..2*Pi], [t^2, t^3, t=-1.5..1.5]], linestyle=[14], scaling=constrained);
```



Анимации – синтактични правила

Maple позволява анимиране на графична информация. Maple-анимацията е последователност от детайлни екрани, които се визуализират бързо, като кадри на кино.

Има две команди за анимиране в Maple:

animate

animate(3d)

Тези инструкции са определени в пакета *plots*.

Анимация 2D

Двумерната анимация се задава по следните синтактични правила:

animate (y- term, x=range, time=range, option);

Всяка анимация има стандартно 16 кадъра. С добавяне на опцията **frames= number of frames** анимацията може да стане по-фина.

Пример 1 за 2D анимация

[> with(plots):

[> animate(sin(x*t), x=- 10..10, t=1..2,
frames=50);

=>прозорец на графиката

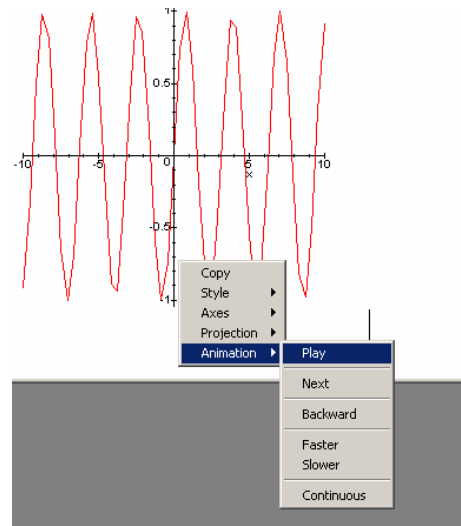
=>кликване с десен бутон на мишката

=>падащо меню

=>избираме ‚Animation‘

=> подменю Animation

=>стартираме анимацията с ‚play‘



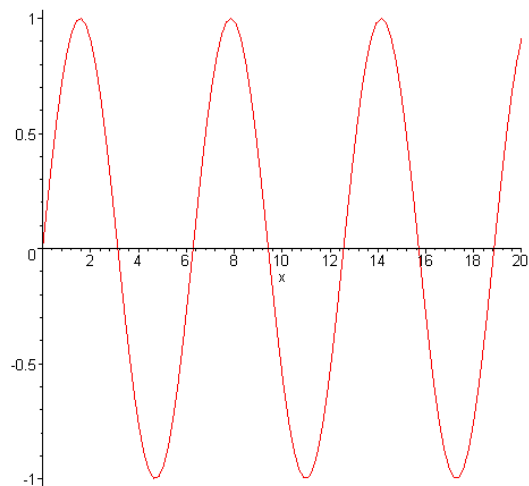
Пример 2 за 2D анимация

Създаване на анимация с 2d графики на функции.

[> with(plots):

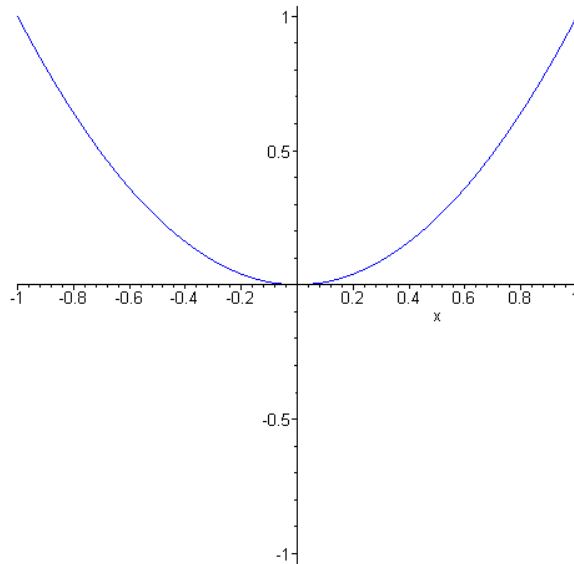
[> animate(sin(x-t), x=0..20,t=0..8*Pi,
frames=50);

50 детайлни кадъра



Пример 3 за 2D анимация

```
[> animate(x^2*cos(t) , x=-1..1 , t=0..8*Pi , color=blue, frames=50);
```



Анимация 3D - синтаксис

Синтактичните правила за тримерната анимация са подобни на тези за двумерната:

animate (z-term, y- range, x=range, time=range, option);

Стандартно тримерната анимация има 8 кадъра. С опцията **frames= number of frames** е лесно този брой да се промени.

Пример за 3D анимация

```
[> with(plots):
```

```
[> animate3d([s*sqrt(time), t-time, 2*s*cos(t*time)], s=1..3, t=1..4, time=2..4);
```

