

## Programové prostředí


Program se skládá z několika různých typů řádků.

1. Řádek pro komentář = program jej neprovede.

Vložíte jej stisknutím CTRL+T, přes menu Insert -> Text, kliknutím na .



Řádek začíná [

2. Řádek s informací pro program = program jej provede.

Vložíte jej stisknutím CTRL+M, přes menu Insert -> Maple Input, kliknutím na , tento typ je implicitní, pokud byste potřebovali i dát i sem komentář, vložte znak # a všechno za ním je komentář

Řádek začíná [>

3. Sekce, ta může obsahovat řádky s komentáři i se vstupy.

Sekce začíná  a končí . Může obsahovat podsekce, podsekce podsekcí, .... Lze ji sbalit (tj. není vidět obsah) a rozbalit (tj. je vidět obsah).


### Jak jednoduše zvětšit písma v programové ploše?

Nahoře jsou tři lupy, vlevo je nejmenší, vpravo je největší.



### Na co je potřeba si dát pozor.

V červeném Maplu pozor na mezery, zvlášť u indexů. Mezera je malá a tak si jí nevšimnete. Maple pak nepočítá.

Konec zadané informace oznamujete programu buď : (nepotřebuji vidět, „co to udělalo“) nebo ; (potřebuji vidět, „co to udělalo“). Na jednom řádku může být víc zadaných informací. Zadání provedu stlačením ENTER, nebo kliknutím na jeden ze symbolů . První provede aktuální řádek, druhý provede CELÝ program.

> **a=3 ; b=2 : c=5 ;**

$a = 3$

$c = 5$

U červeného Maplu není potřeba dávat na konec řádku ; nebo :

### Rozdíl mezi := a =

Pokud zadáte „jen“ = říkáte, že je nějaká rovnice, kdy levá strana = pravé, ale rovnici nepojmenujete a tedy o ní program nic neví

> **A=1 ; B=2 ; C=A+B ;**

$A = 1$

$$B=2$$
$$C=A+B$$

Dotaz na obsah proměnných A a B

> **A;B;**

A  
B

Pokud zadáte := , tak program pravou stranu výrazu přiřadí levé straně (na levé straně je proměnná). Pokud lze výraz vyčíslit, tak jej program vyčíslí.

> **A:=3;B:=4;C:=A+B;**

A:=3  
B:=4  
C:=7

> **A;B;**

3  
4

Pokud chci programu říct, že existuje rovnice jménem rovnice, jejíž levá strana je  $D+2*C$  a pravá strana je  $A+B$  a chci ji vyřešit, tak to provedu takto:

> **rovnice:=D+2\*C=A+B;D;solve(rovnice);D;**

*rovnice:=D+14=7*  
D  
-7  
D

Jak je poznat z posledního příkazu, D; program sice spočítal, že D je -7, ale nedostal za úkol si to zapamatovat. Příkaz solve vyřeší rovnici (nebo soustavu rovnic), pokud je v rovnici jen jedna neznáma, nemusím programu říkat, kterou neznámou z rovnice chci spočítat.

**Je rozdíl mezi malými a velkými písmeny.**

> **a:=3;A:=5;a+A;**

a:=3  
A:=5  
8

**Program si pamatuje i „smazané informace“.**

Tak to může vypadat, ale je i není to pravda. Pokud programu zadám, že A:=5: Dám ENTER, tak si to program pamatuje. Když tento řádek smažu, není důvod, aby to

program zapomněl, ale uloží-li tento soubor a ukončím program Maple, tak po opětovném načtení samozřejmě neví, že A je 5. Pokud chci, aby program všechno, co bylo zadané, zapomněl, použiji příkaz restart: (nebo restart;) je to jedno, protože tento příkaz nemá žádný výstup. Je vhodné jej napsat hned jako první, pokud pustíte program znovu od začátku, tak program Maple nebude maten minulými výsledky.

> **restart:**

Pokud chci, aby program zapomněl jen určitou proměnnou (třeba x), řeknu mu to takto:

> **x:='x' ;**

>

*x := x*

nebo

> **unassign('x') ;**

### Program si vymýšlí nezadané informace.

Pokud máte otevřené 2 a víc souborů a pracujete v nich, jsou informace z obou ukládány do paměti a používány ve VŠECH souborech nezávisle na tom, ve kterém byly použity.



Nejjednodušší je nepouštět 2 a více souborů najednou. Klidně můžete mít otevřeno více souborů, ale ENTER, a symboly používejte jen v jednom.

### Co znamenají různé závorky? () [] {}

Závorky () jsou „normální“. Např.  $\sin(1)$ .

Závorky [] se používají u vektorů a matic. Např.  $A[1,2]$  znamená prvek z matice A, může to být číslo, výraz i rovnice.

Závorky {} se používají, např. když zadáváte parametry, aby program poznal, které k čemu patří. Např. `vysledek:=solve({x+y=5,x-y=3},{x,y});` takto chci spočítat soustavu dvou rovnic o dvou neznámých a výsledek uložit do proměnné vysledek. Pokud budu chtít, aby si program zapamatoval, kolik je x a kolik y, tak mu to řeknu příkazem `assign(vysledek);`

> **vysledek:=solve({x+y=5,x-y=3},{x,y});**

*vysledek:= {x=4,y=1}*

> **x;y;assign(vysledek);x;y;#x a y zna, az po prikazu assign**

*x*

*y*

*4*

*1*

### Počítá mi to divně, pokud používám desetinnou čárku.

Nepočítá, pro něj to znamená komplexní číslo. Ten druhý výraz znamená  $(3+4*i) - (1+5*i)$  a to je  $2-1*i$

```
> 3,4-1,5 ; (3,4) - (1,5) ;
```

3,3,5

2, -1

### sin, cos, tan

Argumenty v goniometrických funkcích používá program v radiánech, tedy  $\sin(90)$  není 1., ale jedna je  $\sin(\pi/2)$ .  $\pi$  je Ludolfovo číslo  $\pi$ .

```
> sin(90) ; sin(Pi/2) ;
```

$\sin(90)$

1

### Jak se liší $\sin(90)$ a $\sin(90.)$ a $\sin(90.0)$ ?

U toho prvního ho program vyčíslí, jen pokud je to PŘESNÁ hodnota, u zbylých dvou vždy, pokud tam ještě není konstanta. Pokud před . není číslice, tak program doplní 0, tedy

```
> sin(90) ; sin(90.) ;
```

$\sin(90)$

0.8939966636

### Kolik je vlastně $\pi$ ?

Zadáním  $\pi$ , Vám program odpoví přesně  $\pi$ . Trik  $1.0*\pi$  Vám také nepomůže, protože sice vyčíslí „nepřesně“ 1.0, ale  $\pi$  je zase  $\pi$ . Dát za něj tečku je špatně. Ale můžete programu říct, že hodnotu  $\pi$  vyčíslit. To se provede příkazem `evalf( $\pi$ )`;

```
> Pi ; 1.0*Pi ; evalf(Pi) ;
```

$\pi$

$1.0 \pi$

3.141592654

### Jak dostanu nápovědu třeba k příkazu `evalf`? A jaké jsou VŠECHNY příkazy?

Zadáte za otazník příkaz, který Vás zajímá a zadáte ENTER (bez ; nebo :), nápověda se otevře v novém okně. Samotný ? Vám otevře nápovědu, kde se dá vše najít - doklikať.

```
> ?evalf
```

```
> ?
```

## Zakázané proměnné.

Do některých proměnných nemůžete zapsat, protože se jedná o konstanty nebo příkazy, např. Re, D, Pi. pi konstanta není a můžete si ji tedy definovat podle libosti.

```
> Re:=5;D:=5;Pi:=3.14;pi:=3.14159;
```

```
Error, attempting to assign to `Re` which is protected. Try declaring `local Re`; see ?protect for details.
```

```
Error, attempting to assign to `D` which is protected. Try declaring `local D`; see ?protect for details.
```

```
Error, attempting to assign to `Pi` which is protected. Try declaring `local Pi`; see ?protect for details.
```

$$\pi := 3.14159$$

## Řecká písmena.

Píší se podle anglické ho pravopisu, tedy  $\alpha$  získáte zadáním alpha,  $\gamma$  zadáním gamma,  $\Gamma$  zadáním Gamma. Pokud je první písmeno v názvu velké, jedná se o velké řecké písmeno.

Seznam získáte dotazem ?greek

V červeném Maplu lze řecká písmena zadat i kliknutím na odpovídající písmeno v okně vlevo.

## Řešení rovnic, když mi zajímají jen některé kořeny. Řešení nelineárních rovnic.

```
> rovnice:=0=x^3+2*x^2+x+1;solve(rovnice);
```

$$\begin{aligned} & \text{rovnice}:=0=x^3+2x^2+x+1 \\ & -\frac{(100+12\sqrt{69})^{(1/3)}}{6} - \frac{2}{3(100+12\sqrt{69})^{(1/3)}} - \frac{2}{3}, \frac{2(100+12\sqrt{69})^{(1/3)}}{12} \\ & + \frac{1}{3(100+12\sqrt{69})^{(1/3)}} - \frac{2}{3} \\ & + \frac{1}{2}I\sqrt{3} \left( -\frac{(100+12\sqrt{69})^{(1/3)}}{6} + \frac{2}{3(100+12\sqrt{69})^{(1/3)}} \right), \frac{(100+12\sqrt{69})^{(1/3)}}{12} \\ & + \frac{1}{3(100+12\sqrt{69})^{(1/3)}} - \frac{2}{3} \\ & - \frac{1}{2}I\sqrt{3} \left( -\frac{(100+12\sqrt{69})^{(1/3)}}{6} + \frac{2}{3(100+12\sqrt{69})^{(1/3)}} \right) \end{aligned}$$

Pokud mi zajímá řešení jako číslo.

```
> fsolve(rovnice);
```

-1.754877666

Ale program mi řekne JEN jedno. Já chci i ty komplexní.

```
> fsolve(rovnice, complex);  
-1.75487766624669 -0.122561166876654-0.744861766619744I,  
-0.122561166876654+0.744861766619744I
```

Některé rovnice program nevyřeší analyticky (solve), ale jen numericky, (fsolve).

```
> solve(ln(x)=cos(x));
```

RootOf( $_Z - e^{\cos(_Z)}$ )

```
> fsolve(ln(x)=cos(x));
```

1.302964001

Co když mi zajímá JEN kladný kořen?

Tak to programem řeknu parametrem v příkazu fsolve, v jakém rozmezí má být neznámá. Např. chci, aby z bylo od -1 do 5 mu řeknu z=-1...5 těch teček musí být víc než 2, klidně i 50, ale 3 až 5 je tak akorát.

```
> rovnice:=x*x-1*x-6=0; solve(rovnice);  
fsolve(rovnice,x=0...infinity); # infinity je nekonecno,  
zalomeni provedeno pomoci TAB
```

$rovnice:=x^2 - x - 6 = 0$

3, -2

3.000000000

### Záleží v Maplu na pořadí?

Ano i ne.

Zde na pořadí záleží:

Pokud zadáte:

```
> a:=5;b:=6;c:=a+b;
```

$a := 5$

$b := 6$

$c := 11$

Je  $c = 11$ , pokud  $c$  nezměníte, změna  $a$  nebo  $b$  už nemá na  $c$  vliv.

```
> a:=6;c;
```

$a := 6$

11

Stejně, jako když vyřešíte rovnici a výsledek napíšete do proměnné, pak je proměnná hodnota  $a$  změna v rovnici její hodnotu už neovlivní.

Pokud zadáte:

```
> Z:=X+Y;X:=5;Y:=6;Z;
```

```
Z:=X+Y
```

```
X:=5
```

```
Y:=6
```

```
11
```

Je Z stále  $X + Y$ , pokud změníte X nebo Y, změní se AUTOMATICKY hodnota Z, protože Z je STÁLE  $X + Y$ .

```
> X:=6;Z;
```

```
X:=6
```

```
12
```

Když zadáte nejdřív b nebo Y, až potom a nebo X, tak to vliv nemá.

**V cizím programu vidím divnou věc.**

```
> 1e5-1e4;
```

```
90000.
```

eČÍSLO je totéž, jako \*10 umocněné na ČÍSLO, tedy 100000 – 10000;

```
> ln(5)-log(5);
```

```
0
```

$\ln(5)-\log(5)$  je nula! ANO, protože i log je přirozený logaritmus, dekadický je  $\log_{10}$

```
> ln(5.)-log10(5.);#bez teček za petkami by to Maple nevyčíslil
```

```
0.9104679077
```

**Co mám dělat?**

Občas ukládat CTRL+S, zvlášť, před příkazy solve, fsolve, simplify, u kterých nejvíc hrozí zamrznutí počítače.

Rozumně si pojmenovávat proměnné, např. a, b, c jsou sice krátká jména, ale co popisují?

d průměr, d\_a průměr s dolním indexem a, V objem, v rychlost a podobně. Rey nebo re může být Reynoldsovo kritérium. Do komentáře psát jednotky, ZVLÁŠŤ pokud nejsou základní (mm). Případně, co ve které proměnné je. Teď to víte, ale za půl roku?

Rozumně si pojmenovávat rovnice. Jméno rovnice není ani krátké a ani nepopisuje, o co jde.

Třeba br\_a může znamenat bilanční rovnici pro látku a, nebo br1a může být bilanční rovnice pro látku a v prvním uzlu, s5 zase popisovat, že součet molárních zlomků v proudu 5 je 1.

První řádek obsahuje restart: # Kdo vytvořil tento program a co má dělat (stačí i číslo úlohy).

To, že něco vyřeším, pro systém nic neznamená, pokud výsledek nezapišu tam, kam patří.

Když integruji podle konverze, těžko mu můžu napsat, že konverze je 90%. To musí být dvě proměnné, podle jedné integruji a ta druhá může být konverze\_k (jako konverze na konci, ta pak bude 0,9 tj. těch zmiňovaných 90%).

Výsledek řešení dát do proměnné vysledek a pokud je „reálný“, pak ho z té proměnné dát kam patří funkci assign. Pokud reálný není, tak najít jiný omezením rozsahu neznámé (neznámých).

## Co nemám dělat?

Nepropadejte panice. Pokud Vám to nedělá, to co má. Vypište si z řádku, který zlobí, všechny neznámé, jestli jsou skutečně neznámé (pokud má proměnná a hodnotu 5, tak její hodnotu z rovnice nevyřešíte), a jestli jsou všechny známé skutečně známé.

Používat písmena s háčky a čárkami ani u proměnných a raději ani u popisu. Když bude mít někdo jiné fonty, může být pro něj text nečitelný.

Nedávat do názvu proměnných znaky jako # <>.:;.! ? +-\*/^ mezera a podobné.

Pokud mi stačí numerické řešení navíc mi zajímají jen některé kořeny, tak proč programu nezjednodušit život -> reseni:=fsolve({br3a,br3},{x3a,m5},x3a=0....1, m5=0....99e99);

Nepsat rovnice hezky užíváním různých druhů závorek, každá má svůj konkrétní význam!

Raději v červeném Maplu nepoužívejte indexy, nebo aspoň, dokud „to“ neodladíte. Když je v indexu la nebo l\_a nebo la tak to skutečně nepoznáte.

Často u reaktorů často integrujete výraz 1/(1-x) pokud neřeknete jinak, Maple předpokládá x od 1 do nekonečna a výsledkem je tedy -ln(x-1), místo předpokládaného -ln(1-x), protože argument má být v absolutní hodnotě, jedná se o stejný výsledek, ale reálný je vždycky jen jeden, druhý je logaritmus záporného čísla.

## Jednotky

V „červeném“ Maplu jde zadávat jednotky a program s nimi dál pracuje. Pokud budete potřebovat spočítat dráhu, znáte rychlost a potřebnou dobu a obě hodnoty zadáte s jednotkami, program Vám spočítá velikost dráhy včetně jednotek.

$$> v := \frac{5.0[m]}{[s]}$$

$$v := \frac{5.0 [m]}{[s]}$$

$$> \tau := 3.0[s]$$

$$\tau := 3.0 [s]$$

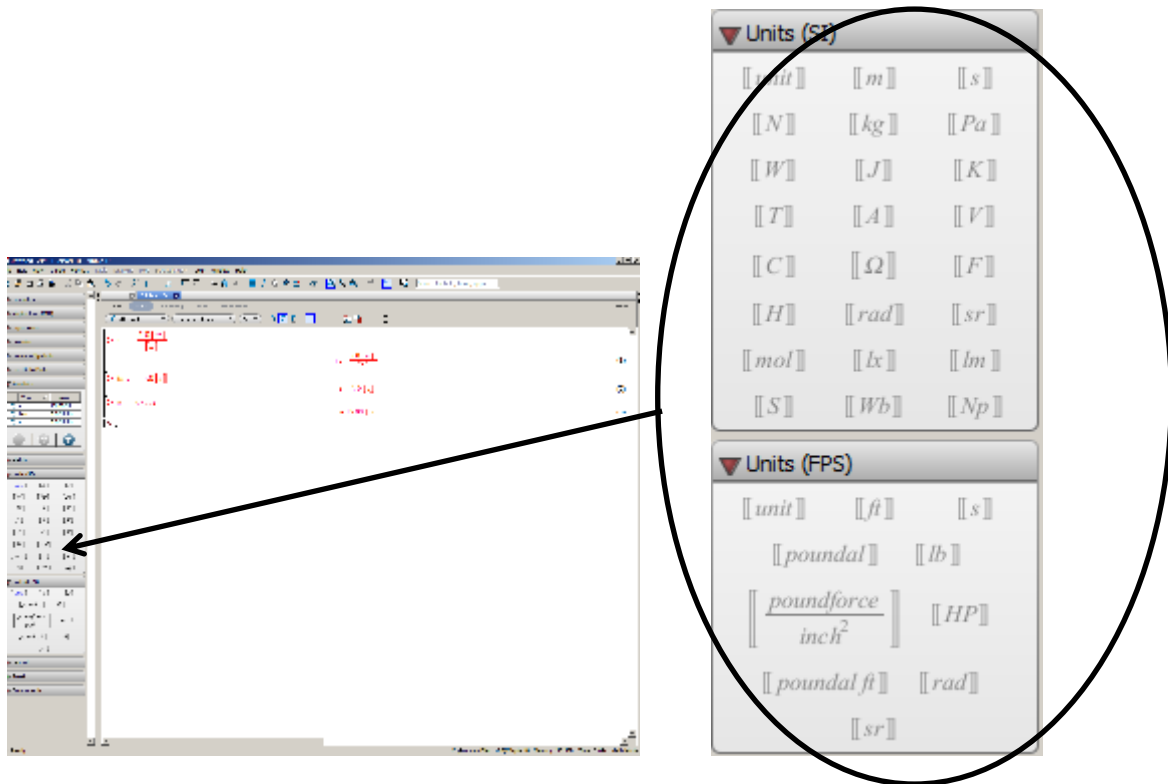
$$> s := v \cdot \tau$$

$$s := 15.00 [m]$$

Zlomky program udělá sám, jakmile stisknete klávesu / , jednotky se nepíší pomocí hranatých závorek, ale doklikávají se z okna Units, které je vlevo. Řádky není potřeba zakončovat : nebo ; , ale když „to tam dáte“, tak to ničemu nevadí.



POZOR: Pa je totéž, jako  $\text{kg}/(\text{m s s})$ . J je totéž, jako  $\text{kg m m} / (\text{s s})$



### Příklad: Tři způsoby řešení bilanční úlohy.

Houby obsahují 92 % vody. Původně vážily 5 kg, po usušení váží 2,2 kg. Kolik % vody se obsahují po usušení? Kolik kg vody se odpařilo?

Vodu označím jako a, sušinu jako b. Před sušením bude 1, po sušení bude 2 a odpařená voda 3. Požadované neznámé jsou  $m_3$  a  $w_{2a}$ .

1. Maple jako chytrější kalkulačka. Všechno musím vymyslet sám, Maple to jen spočítá.

```
> restart;
> m1:=5;m2:=2.2;m3:=m1-m2;
      m1 := 5
      m2 := 2.2
      m3 := 2.8

> w1a:=0.92;w3a:=1;
      w1a := 0.92
      w3a := 1

> w2a:=(w1a*m1-w3a*m3)/m2;
      w2a := 0.8181818182
```

2. Maple jako řešitel rovnic. Sestavím rovnice a Maple je vyřeší.

```
> restart;
> m1:=5;m2:=2.2;w1a:=0.92;w3a:=1;
      m1 := 5
      m2 := 2.2
      w1a := 0.92
      w3a := 1
```

```

> br:=m1=m2+m3;
                                     br := 5 = 2.2 + m3
> bra:=w1a*m1=w2a*m2+w3a*m3;
                                     bra := 4.60 = m3 + 2.2 w2a
> reseni:=solve({br,bra},{m3,w2a});
                                     reseni:= {m3=2.800000000, w2a=0.8181818182}
> assign(reseni);

```

Tato varianta je přehlednější a jednodušší, proto ji doporučuji.

Značím: br – bilanční rovnici, bra – bilanční rovnici pro složku a, pokud by bylo více uzlů, pak doplním ještě číslo uzlu.

3. Tato úloha lze řešit pomocí vektorů a matice.

```

> restart;
> with(LinearAlgebra);
> m:= <-m1|m2|m3>;
                                     m := [-m1, m2, m3]
> w:=<<w1a, w2a, w3a>|<w1b, w2b, w3b>|<1, 1, 1>>;
                                     w :=  $\begin{bmatrix} w1a & w1b & 1 \\ w2a & w2b & 1 \\ w3a & w3b & 1 \end{bmatrix}$ 
> v:=VectorMatrixMultiply(m,w);
                                     v := [-m1 w1a + m2 w2a + m3 w3a, -m1 w1b + m2 w2b + m3 w3b, -m1 + m2 + m3]
> m1:=5;m2:=2.2;w1a:=0.92;w3a:=1;
                                     m1 := 5
                                     m2 := 2.2
                                     w1a := 0.92
                                     w3a := 1
> reseni:=solve({v[1],v[3]},{m3,w2a});
                                     reseni:= {m3=2.800000000, w2a=0.8181818182}
> assign(reseni);

```

m je vektor obsahující celkové hmotnosti proudů na jedné straně - na druhé straně (proto je před m1 znaménko mínus)

w je matice obsahující složení proudů, poslední sloupec je pro celkovou bilanci

v je vektor, jehož každý prvek obsahuje bilanci jedné složky, poslední obsahuje celkovou bilanci, v[1] je první prvek z vektoru, zde je to bilance složky a, protože je od výstupu odečten vstup, musí být hodnota každého prvku vektoru v nula.

V příkazu solve obsahuje rovnice jen jednu stranu, pokud je to tak, tak systém předpokládá, že je druhá strana = 0. **reseni := solve ({v[1]=0, v[3]=0}, {m3, w2a});** je totéž.

Pokud bych chtěl spočítat w1b, w2b a w3b, použil bych dodatečné vztahy, že součet hmotnostních zlomků všech složek v proudu je 1.

### Potřebujete spočítat konstanty pro proložení bodů obecnou funkcí?

Lineární a polynommické Vám spočítá kdejaký program, ale obecné Maple.

Máte sadu dvojic [x,y] a obecnou funkci. Zde je pět dvojic x,y a funkce má tvar  $y = b + c/x$ .

Pomocí metody nejmenších čtverců hledáte správné hodnoty b a c.

Zapsáno v Maplu.

> **with(CurveFitting):**

>

**x1:=1.5:x2:=2.5:x3:=4:x4:=5.5:x5:=7.2:y1:=3.5:y2:=5.5:y3:=6:y4:=5.0:y5:=3.2:**

> **LeastSquares([x1,x2,x3,x4,x5],[y1,y2,y3,y4,y5],x,curve=b+c/x);**

$$5.01657361404112 - \frac{1.14993176401827}{x}$$

Proměnná b je 5.01657361404112 a c je 1.14993176401827.

### Integrace, analytické a numerické řešení.

Při výpočtu času potřebného pro zadanou konverzi často vychází integrály typu

$$\frac{1}{(-ca_0 \zeta + ca_0)(-ca_0 \zeta + cb_0)}$$

Analytické řešení integrálu se počítá takto:

>> **int(1/((ca0-ca0\*zeta)\*(-ca0\*zeta+cb0)),zeta);**

$$-\frac{\ln(ca_0 \zeta - cb_0)}{ca_0(-cb_0 + ca_0)} + \frac{\ln(-1 + \zeta)}{ca_0(-cb_0 + ca_0)}$$

Ale POZOR, integrál z  $(1/(1-x))dx$  NENÍ  $\ln(1-x)$ , ale  $\ln|1-x|$ , tj.  $\ln$  absolutní hodnoty. Tak čitatel prvního zlomku není  $\ln(ca_0 \cdot \zeta - cb_0)$ , ale  $\ln|cb_0 - ca_0 \cdot \zeta|$ . To popisuje, kolik zůstalo latky B na konci. Druhého zlomku není  $\ln(-1+\zeta)$ , ale je  $\ln|1-\zeta|$ . To popisuje, kolik zůstalo látky A na konci ( $ca_0$  bylo vytknuto). Maple má občas na výběr a může nabídnout sice správné, ale fyzikálně nerealizovatelné řešení.

Pokud Maplu neřeknete, že  $0 < \zeta < 1$ , tak předpokládá, že  $\zeta > 1$ , proto ty výsledky.

Numerické řešení téhož integrálu:

```
> ca0:=22.4:cb0:=55.2:zetak:=0.99:int(1/((ca0-ca0*zeta)*(cb0-  
ca0*zeta)),zeta=0...zetak);  
0.005568708899
```

Jsou tam dosazeny počáteční koncentrace a konečná konverze (zetak).

POZOR! Pokud bych v integrálu nahradil zetak za zeta a řekl mu, že zeta = 0.99 a chtěl spočítat integrál, ohlásí chybu.

Funkci int lze dát i do řešitele.

```
> solve(2=int(5-x,x=-1...xk));  
5+4√2,5-4√2
```

V tomto případě je jediná neznámá a to konečná mez proměnné.

## Derivace

Maple umí také derivovat.

```
> diff(a*x^3+b*x^2+c*x+d+e/x+f/(x*x)+exp(2*x)+ln(x),x);  
3ax2+2bx+c- $\frac{e}{x^2}$ - $\frac{2f}{x^3}$ +2e(2x)+ $\frac{1}{x}$ 
```

POZOR Diff je jiná funkce. Řešitel diferenciálních rovnic se jmenuje dsolve.

## Vytvoření vlastní funkce

Definuji funkci „funkce“  $x*x-x+2$ , tak, abych zadal x a dozvěděl se výsledek.

```
> funkce:=x->x*x-x+2;  
funkce := x → x x - x + 2  
  
> funkce(1); funkce(3);  
2  
8
```

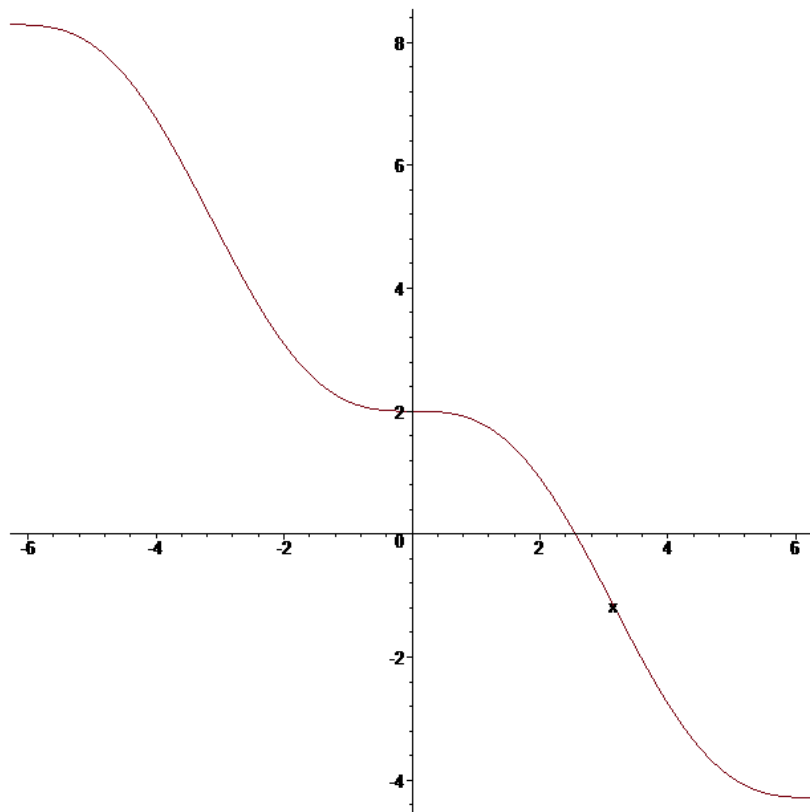
A jde to i pro více argumentů.

```
> funkce2:=(x,y)->x-y;  
funkce2 := (x, y) → x - y  
  
> funkce2(2,3); funkce2(3,2);  
-1  
1
```

## Grafy – zobrazení funkcí

Jak vypadá graf funkce  $y=\sin(x)-x+2$  ?

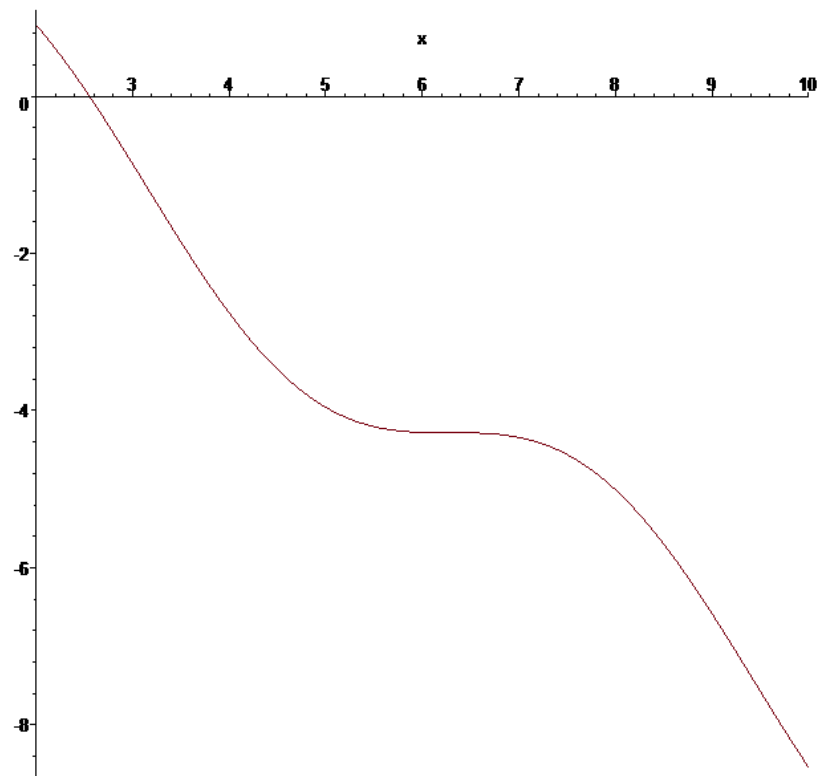
```
> plot(sin(x) - x + 2);
```



Samozřejmě, že  $x$  je v radiánech.

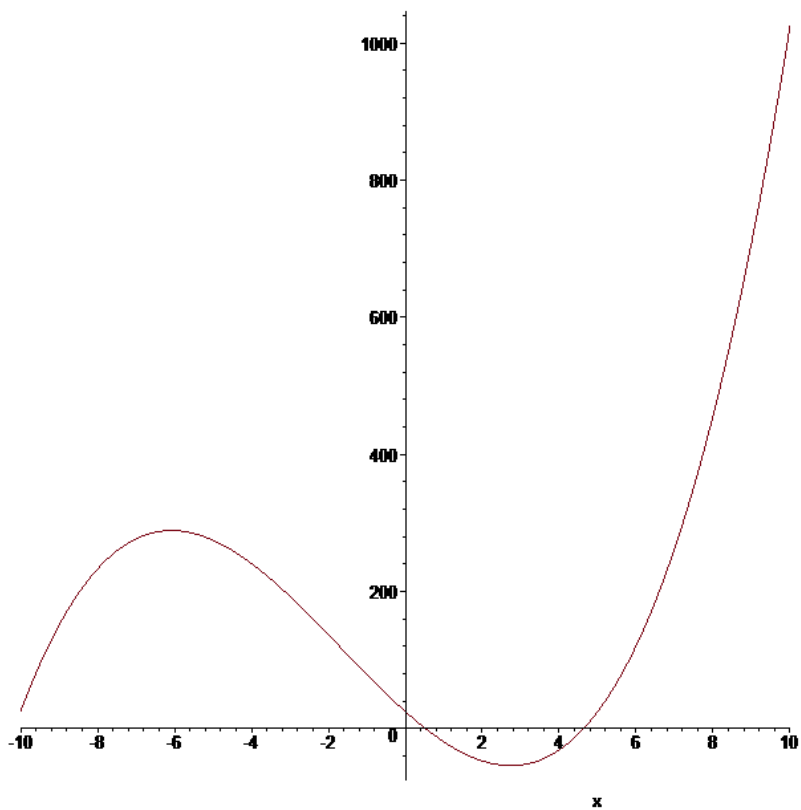
A co když mi ten graf zajímá pro  $2 < x < 10$ ?

```
> plot(sin(x) - x + 2, x=2..10);
```



A může se mi to hodit třeba k tomu, abych zjistil, pro jaké  $x$  je řešení  $y = 0$  a jestli těch řešení není víc.

> `plot(x*x*x+5*x*x-50*x+25);`



Jeden kořen je asi -11, druhý asi 0.4 a třetí asi 4.5. Je to šikovné hlavně pro složité funkce, abych se ujistil, že jsem na nějaký kořen nezapomněl.