

POČÍTAČOVÁ PODPORA V ELEKTROTECHNICE

ING. LENKA ŠROUBOVÁ, PH.D.
lsroubov@kte.zcu.cz

ING. PETR KROPÍK, PH.D.
pkropik@kte.zcu.cz

KATEDRA TEORETICKÉ ELEKTROTECHNIKY
FAKULTA ELEKTROTECHNICKÁ
ZÁPADOČESKÁ UNIVERZITA V PLZNI

MÍSTNOST: EK602



Funkce pro vstup a výstup

Formátovaný textový výstup

`fprintf`

Příklad:

```
m = 123.3456;  
fprintf('%f\n', m);  
123.345600  
fprintf('%e\n', m);  
1.233456e+002  
fprintf('%E\n', m);  
1.23456E+002  
fprintf('%g\n', m);  
123.346  
fprintf('%G\n', m);  
123.346
```

```
n = 1.23440000e34;  
fprintf('%f\n', n);  
123440000000000001000000000000000000.000000  
fprintf('%e\n', n);  
1.234400e+034  
fprintf('%E\n', n);  
1.234400E+034  
fprintf('%g\n', n);  
1.2344e+034  
fprintf('%G\n', n);  
1.2344E+034
```

Funkce pro vstup a výstup

Pozn.: porovnání s **format**, který nastavuje výstupní formát zobrazení čísla na obrazovku, např.

```
y = 123.4567890123456789;  
format long e  
y  
y = 1.234567890123457e+002  
format short e  
y  
y = 1.2346e+002  
format long g  
y  
y = 123.456789012346  
format short g  
y  
y = 123.46
```

```
format  
y  
y = 123.4568  
fprintf('%f\n', y)  
123.456789  
fprintf('%e\n', y)  
1.234568e+002  
fprintf('% .4e\n', y)  
1.2346e+002  
fprintf('% .12e\n', y)  
1.234567890123e+002  
fprintf('%g\n', y)  
123.457
```

Příklad: funkce pro výpis podílu pomocí **disp** a **fprintf** různými způsoby, dělenec a dělitel jsou vstupními parametry funkce

```
function vystupyTest1(coDelit,cimDelit)
if(cimDelit==0)
    disp('Nulou nedelim!!!')
    return
end
x = coDelit / cimDelit;
disp(['Vysledek je: ',num2str(x)]);
hlaska = 'Vysledek je: ';
format short
disp(hlaska);
disp(x);
format long g;
disp(hlaska);
disp(x);
format rat;
disp(hlaska);
disp(x);
fprintf('\nVysledek je %.3f\n',x);
end
```

Volání funkce:

```
vystupyTest1(17,3)
```

```
Vysledek je: 5.6667
```

```
Vysledek je:
    5.6667
```

```
Vysledek je:
    5.666666666666667
```

```
Vysledek je:
    17/3
```

```
Vysledek je 5.667
```

Nebo:

```
vystupyTest1(-59,0)
```

```
Nulou nedelim!!!
```

Pokračování příkladu:

```
function vystupyTest2 (coDelit,cimDelit)
if (cimDelit==0)
    disp('Nulou nedelim!!!')
    return
end
x = coDelit / cimDelit;
fprintf('Vysledek je x = %-8.3f\nA to je spravne...\n',x);
fprintf('\nNebo presneji x = %-8.6f\n', x);
fprintf('\nNyni prehledne v tabulce:\n');
fprintf('|%15s|%15s|%15s|\n', 'Delenec', 'Delitel', 'Podil');
fprintf('|%15.3f|%15.3f|%15.3f|\n\n', coDelit,cimDelit,x);
end
```

vystupyTest2(17,3) - volání funkce

Vysledek je x = 5.667

A to je spravne...

Nebo presneji x = 5.666667

Nyni prehledne v tabulce:

	Delenec	Delitel	Podil
	17.000	3.000	5.667

Pokračování příkladu:

```
function vystupyTest3(coDelit,cimDelit)
if(cimDelit==0)
    disp('Nulou nedelim!!!')
    return
end

x = coDelit / cimDelit;

fprintf('\nJeste jednou, tentokrat zarovnano doleva:\n');
fprintf('|%-15s|%-15s|%-15s|\n', 'Delenec', 'Delitel', 'Podil');
fprintf('|%-15.3f|%-15.3f|%-15.3f|\n\n', coDelit,cimDelit,x);
end
```

zarovnat doleva

Volání funkce:

```
vystupyTest3(17,3)
```

Jeste jednou, tentokrat zarovnano doleva:

Delenec	Delitel	Podil	
17.000	3.000	5.667	

Pokračování příkladu:

```
function vystupyTest4(coDelit,cimDelit)
if(cimDelit==0)
    disp('Nulou nedelim!!!')
    return
end

x = coDelit / cimDelit;

fprintf('\nA oddeleno jen tabelatorem:\n');
fprintf('%15s\t%15s\t%15s\n','Delenec', 'Delitel', 'Podil');
fprintf('%15.3f\t%15.3f\t%15.3f\n\n', coDelit,cimDelit,x);
end
```

tabelátor

Volání funkce:

```
vystupyTest4(17,3)
```

A oddeleno jen tabelatorem:

Delenec	Delitel	Podil
17.000	3.000	5.667

Funkce pro vstup a výstup

Vstup a tisk znaků (příp. textového řetězce)

```
zadany_znak = input('Zadejte písmeno: ', 's');
```

Zadejte písmeno: n

```
fprintf('Bylo zadano: %c\n', zadany_znak);
```

Bylo zadano: n

```
zadany_text = input('Zadejte text: ', 's');
```

Zadejte text: Ahoj, jak je?

```
fprintf('Bylo zadano: %s\n', zadany_text);
```

Bylo zadano: Ahoj, jak je?

Pozn.

```
zadany_text = input('Zadejte text: ', 's');
```

Zadejte text: Ano!

```
fprintf('Bylo zadano: %c\n', zadany_text);
```

Bylo zadano: A

Bylo zadano: n

Bylo zadano: o

Bylo zadano: !

Funkce pro vstup a výstup

Vstup a tisk znaků (příp. textového řetězce)

```
zadany_znak = input('Zadejte písmeno: ', 's');
```

Zadejte písmeno: n

```
fprintf('Bylo zadano: %c\n', zadany_znak);
```

Bylo zadano: n

```
zadany_text = input('Zadejte text: ', 's');
```

Zadejte text: Ahoj, jak je?

```
fprintf('Bylo zadano: %s\n', zadany_text);
```

Bylo zadano: Ahoj, jak je?

řetězec

Pozn.

```
zadany_text = input('Zadejte text: ', 's');
```

Zadejte text: Ano!

```
fprintf('Bylo zadano: %c\n', zadany_text);
```

Bylo zadano: A

Bylo zadano: n

Bylo zadano: o

Bylo zadano: !

znak

Funkce pro vstup a výstup

Pozn.: k zadávání znaků

Pokud uživatel zadá více znaků, lze se přebytečných zbavit např. takto:

```
zadany_znak = input('Zadejte písmeno: ', 's');
```

```
Zadejte písmeno: Ahoj, jak je?
```

```
if (length(zadany_znak) > 1)
zadany_znak = zadany_znak(1,1);
end;
```

```
fprintf('Bylo zadáno: %c\n', zadany_znak);
```

```
Bylo zadáno: A
```

Funkce pro vstup a výstup

Pozn.: k zadávání znaků

Pokud uživatel zadá více znaků, lze se přebytečných zbavit např. takto:

```
zadany_znak = input('Zadejte písmeno: ', 's');
```

```
Zadejte písmeno: Ahoj, jak je?
```

```
if (length(zadany_znak) > 1)
zadany_znak = zadany_znak(1,1);
end;
```

```
fprintf('Bylo zadáno: %c\n', zadany_znak);
```

```
Bylo zadáno: A
```

Funkce pro vstup a výstup

Pozn.: k zadávání znaků

Pokud uživatel zadá více znaků, lze se přebytečných zbavit např. takto:

```
zadany_znak = input('Zadejte písmeno: ', 's');
```

```
Zadejte písmeno: Ahoj, jak je?
```

```
if (length(zadany_znak) > 1)
    zadany_znak = zadany_znak(1,1);
end;
```

```
fprintf('Bylo zadano: %c\n', zadany_znak);
```

```
Bylo zadano: A
```

```
fprintf('Bylo zadano: %5c\n', zadany_znak);
```

```
Bylo zadano:      A
```

Funkce pro vstup a výstup

Pozn.: k zadávání znaků

Pokud uživatel zadá více znaků, lze se přebytečných zbavit např. takto:

```
zadany_znak = input('Zadejte písmeno: ', 's');
```

Zadejte písmeno: **A**hoj, jak je?

```
if (length(zadany_znak) > 1)
    zadany_znak = zadany_znak(1,1);
end;
```

```
fprintf('Bylo zadano: %c\n', zadany_znak);
```

Bylo zadano: **A**

Doplněno 4 mezerami,
 $4 + 1 = 5$

```
fprintf('Bylo zadano: %5c\n', zadany_znak);
```

Bylo zadano: **A**

Funkce pro vstup a výstup

Tisk do řetězce

```
r = sprintf('formátovací sekvence', argumenty)
```

– vrací řetězec, do řetězce **r** jsou vypsané argumenty v požadovaném formátu v uvedeném pořadí, stejně jako u **fprintf**, formátovací sekvence jsou také stejné jako u **fprintf**.

Příklady:

```
zadany_text = input('Zadejte text: ', 's');
```

Zadejte text: Ahoj, jak je?

```
hlaska = sprintf('Bylo zadano: %s\n', zadany_text);
```

```
figure
```

```
% prázdné grafické okno
```

```
title(hlaska)
```

```
% titulek grafu z řetězce hlaska
```



Funkce pro vstup a výstup

Tisk do řetězce

```
r = sprintf('formátovací sekvence', argumenty)
```

– vrací řetězec, do řetězce **r** jsou vypsané argumenty v požadovaném formátu v uvedeném pořadí, stejně jako u **fprintf**, formátovací sekvence jsou také stejné jako u **fprintf**.

Příklady:

```
zadany_text = input('Zadejte text: ', 's');
```

```
Zadejte text: Ahoj, jak je?
```

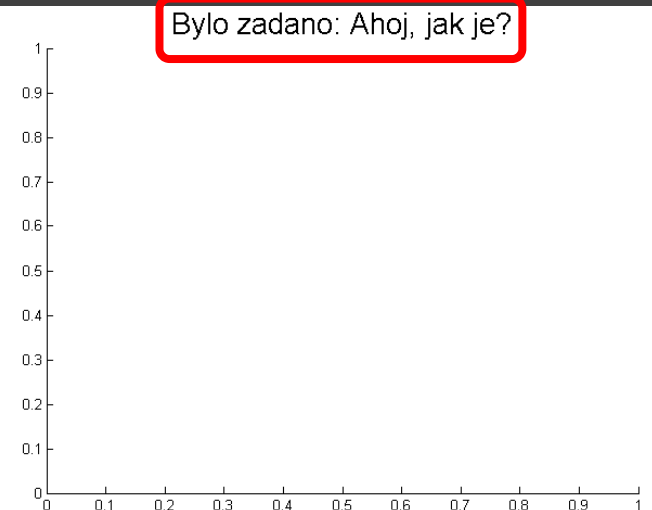
```
hlaska = sprintf('Bylo zadano: %s\n', zadany_text);
```

```
figure
```

```
% prázdné grafické okno
```

```
title(hlaska)
```

```
% titulek grafu z řetězce hlaska
```



Funkce pro vstup a výstup

Tisk do řetězce

```
r = sprintf('formátovací sekvence', argumenty)
```

– vrací řetězec, do řetězce **r** jsou vypsané argumenty v požadovaném formátu v uvedeném pořadí, stejně jako u **fprintf**, formátovací sekvence jsou také stejné jako u **fprintf**.

Příklady:

```
zadany_text = input('Zadejte text: ', 's');
```

```
Zadejte text: Ahoj, jak je?
```

```
hlaska = sprintf('Bylo zadano: %s\n', zadany_text);
```

```
figure
```

```
% prázdné grafické okno
```

```
title(hlaska)
```

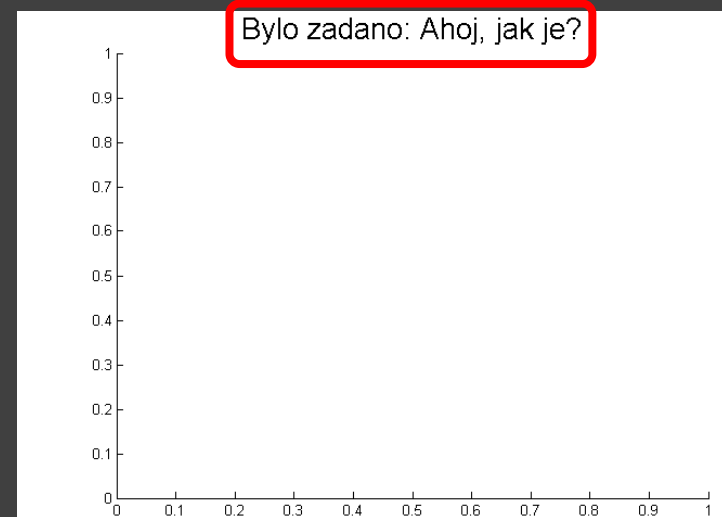
```
% titulek grafu z řetězce hlaska
```

```
disp(hlaska)
```

```
Bylo zadano: Ahoj, jak je?
```

```
fprintf('%s\n', hlaska)
```

```
Bylo zadano: Ahoj, jak je?
```



Funkce pro vstup a výstup

Tisk do řetězce

sprintf

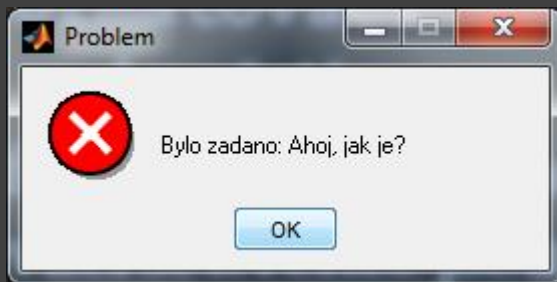
Příklady:

```
zadany_text = input('Zadejte text: ', 's');
```

```
Zadejte text: Ahoj, jak je?
```

```
hlaska = sprintf('Bylo zadano: %s\n', zadany_text);
```

```
msgbox(hlaska, 'Problem', 'error');
```



msgbox (zpráva, titulek, ikona)
- vytvoří okno se **zprávou**, **titulek** se objeví v liště, **ikona** určuje, jaký obrázek ('error', 'help', 'warn') se zobrazí v okně se zprávou.

Pozn.: další okna viz help - pod heslem Predefined Dialog Boxes

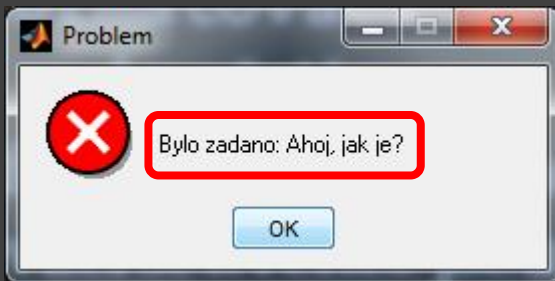
Funkce pro vstup a výstup

Tisk do řetězce

sprintf

Příklady:

```
zadany_text = input('Zadejte text: ', 's');  
Zadejte text: Ahoj, jak je?  
hlaska = sprintf('Bylo zadano: %s\n', zadany_text);  
msgbox(hlaska, 'Problem', 'error');
```



msgbox (zpráva, titulek, ikona)
- vytvoří okno se zprávou, titulek se objeví v liště, ikona určuje, jaký obrázek ('error', 'help', 'warn') se zobrazí v okně se zprávou.

Pozn.: další okna viz help - pod heslem Predefined Dialog Boxes

Funkce pro vstup a výstup

Příklad: Vytvořte uživatelskou funkci **s2cos_graf_s_titulkem** bez parametrů. Tato funkce **s2cos_graf_s_titulkem** bude vykreslovat graf křivky dané rovnicí

$$y = \sin^2(x) \cos(x)$$

pro x z intervalu, jehož jednu mez zadá uživatel z klávesnice, druhá mez je 4π . Je třeba ošetřit, aby první mez nebyla stejná nebo přibližně stejná jako druhá, aby rozdíl mezi nimi byl větší nebo roven 0,025.

Pro výpočet $y = \sin^2(x) \cos(x)$ bude vytvořena další funkce **s2cos** s jedním vstupním parametrem x a jedním výstupním parametrem y . **s2cos** bude volána ve **s2cos_graf_s_titulkem**.

Funkce mohou být napsány každá v samostatném souboru a uloženy ve stejné složce nebo mohou být napsány v jednom souboru (nejprve hlavní funkce **s2cos_graf_s_titulkem**, a potom funkce **s2cos**). Nejdříve vytvoříme funkci **s2cos_graf_s_titulkem**:

```
function s2cos_graf_s_titulkem
horni = 4*pi;
dolni = input('Zadejte pocatecni hodnotu: ');
if ((horni == dolni) || (abs(horni-dolni) < 0.025))
    hlaska = sprintf('Dolni mez %.2f je stejná s horni!', dolni);
    msgbox(hlaska, 'Problem', 'error');
    return;
end
osa_x = linspace(dolni, horni, abs(horni-dolni)*100);
osa_y = s2cos(osa_x);
plot(osa_x, osa_y)
xlabel('x')
ylabel('y')
titulek = sprintf('Graf y = sin(x)^2 cos(x) v mezich od %.2f do
%.2f\n', dolni, horni);
title(titulek)
end
```

```
function y = s2cos(x)
y = (sin(x).^2).*cos(x);
end
```

```

function s2cos_graf_s_titulkem
horni = 4*pi;
dolni = input('Zadejte počáteční hodnotu: ');
if ((horni == dolni) || (abs(horni-dolni) < 0.025))
    hlaska = sprintf('Dolní mez %.2f je stejná s horní! ', dolni);
    msgbox(hlaska, 'Problem', 'error');
    return;
end
osa_x = linspace(dolni, horni, abs(horni-dolni)*100);
osa_y = s2cos(osa_x);
plot(osa_x, osa_y)
xlabel('x')
ylabel('y')
titulek = sprintf('Graf y = sin(x)^2 cos(x) v mezích od %.2f do
%.2f\n', dolni, horni);
title(titulek)
end

```

velikost, absolutní hodnota

je požadován rozdíl
mezi mezemi $\geq 0,025$

volání funkce **s2cos** se
vstupem **osa_x** pro výpočet **y**

```

function y = s2cos(x)
y = (sin(x).^2).*cos(x);
end

```

Funkce pro vstup a výstup

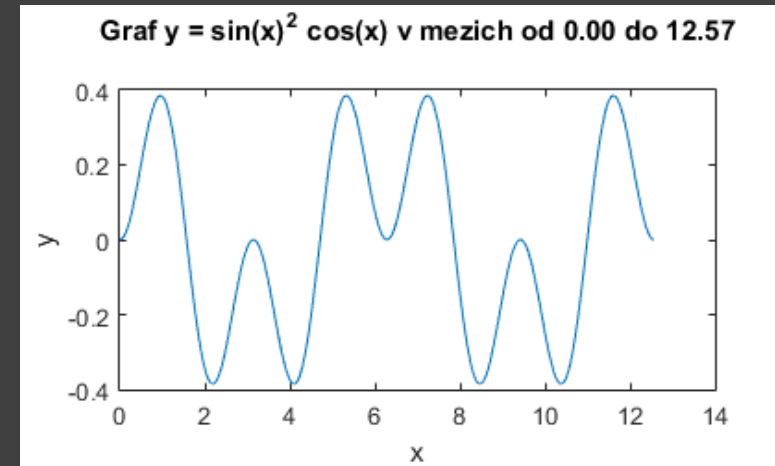
Pokračování příkladu:

Volání funkce jejím názvem:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: 0

a už se vykreslí graf
s příslušným titulkem ...



Funkce pro vstup a výstup

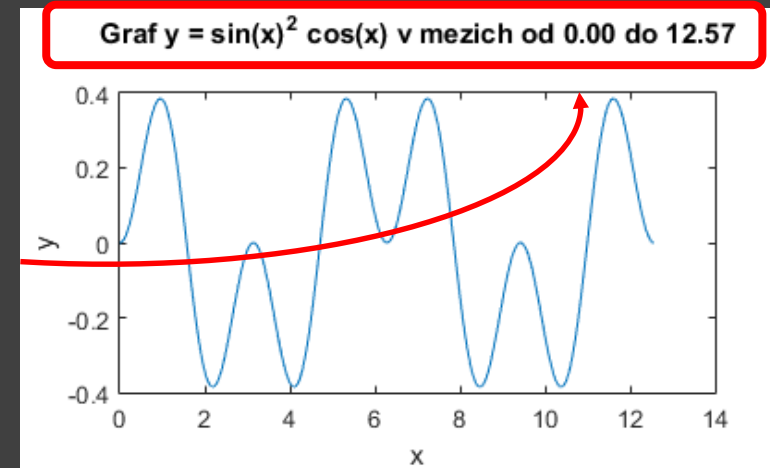
Pokračování příkladu:

Volání funkce jejím názvem:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu:

a už se vykreslí graf
s příslušným titulkem ...



Funkce pro vstup a výstup

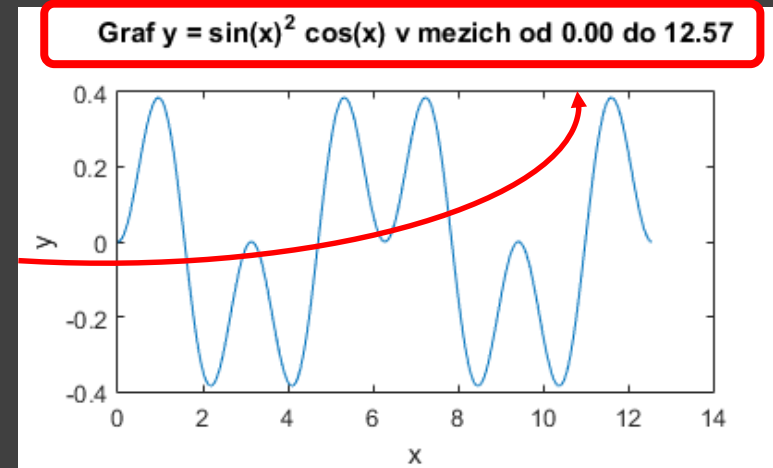
Pokračování příkladu:

Volání funkce jejím názvem:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **0**

a už se vykreslí graf
s příslušným titulkem ...

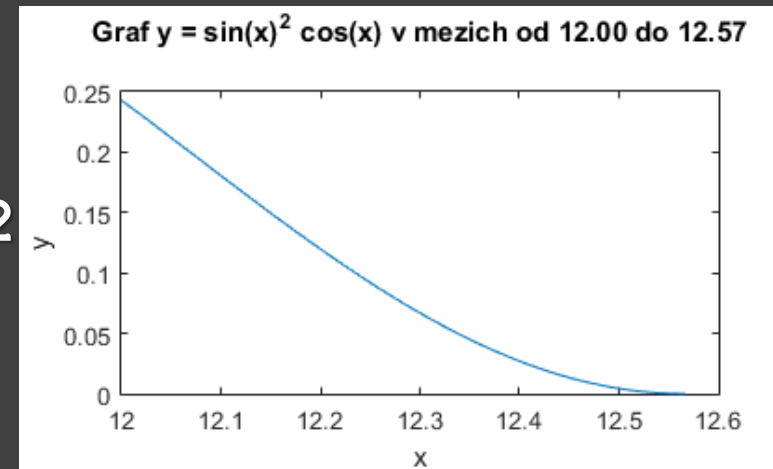


nebo:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **12**

a už se vykreslí graf
s příslušným titulkem ...



Funkce pro vstup a výstup

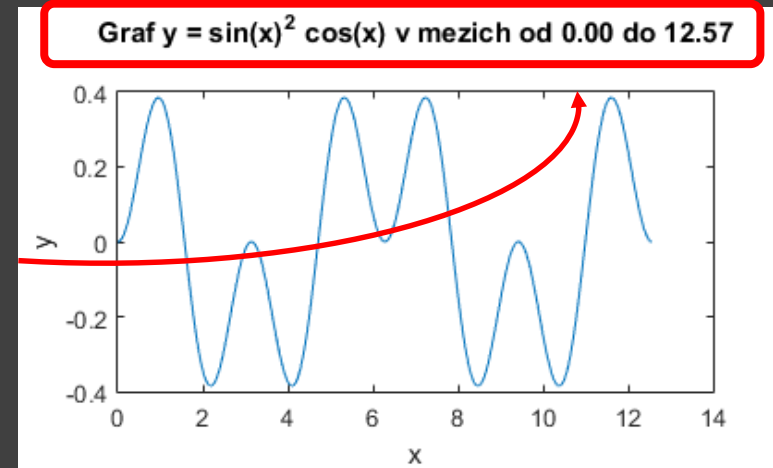
Pokračování příkladu:

Volání funkce jejím názvem:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **0**

a už se vykreslí graf
s příslušným titulkem ...

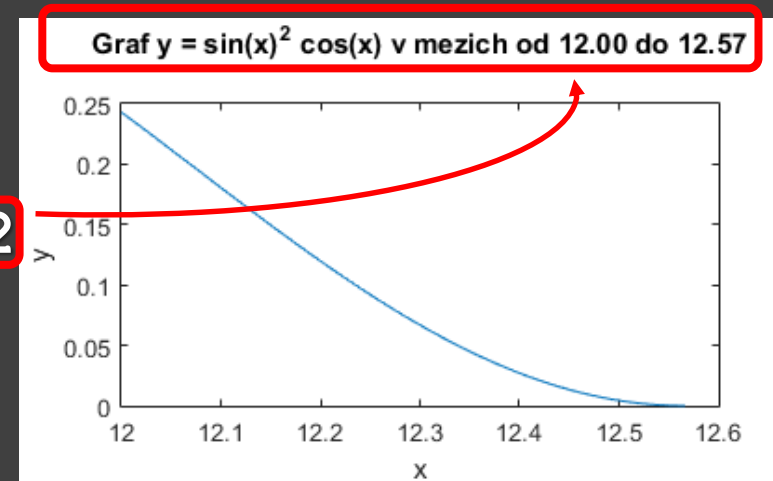


nebo:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **12**

a už se vykreslí graf
s příslušným titulkem ...



Funkce pro vstup a výstup

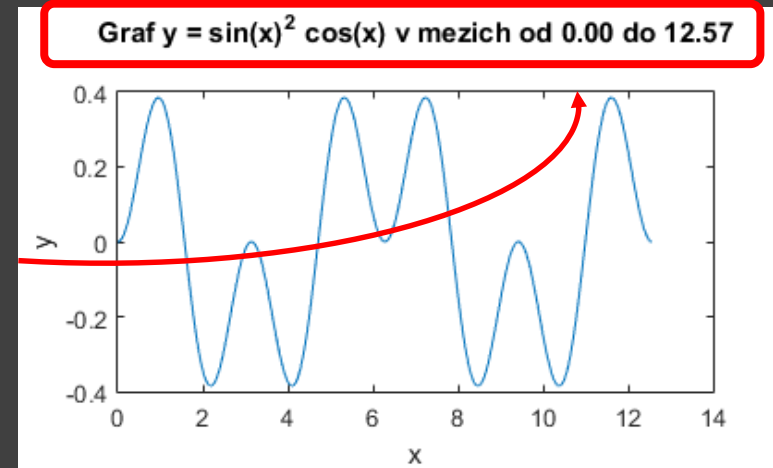
Pokračování příkladu:

Volání funkce jejím názvem:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **0**

a už se vykreslí graf
s příslušným titulkem ...

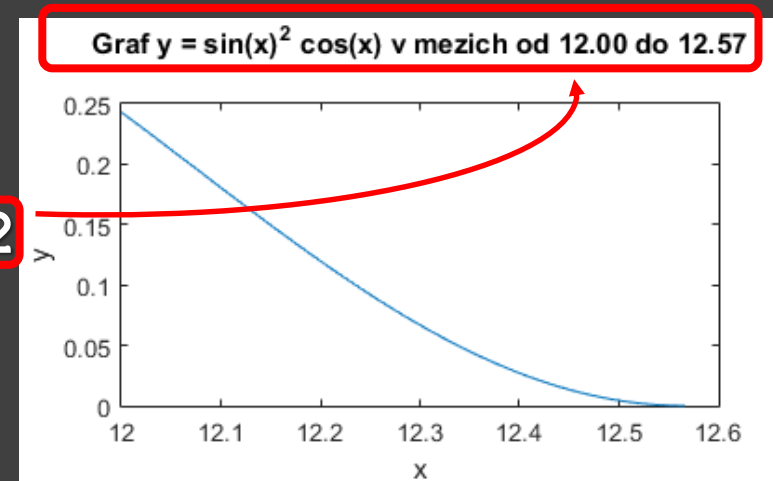


nebo:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **12**

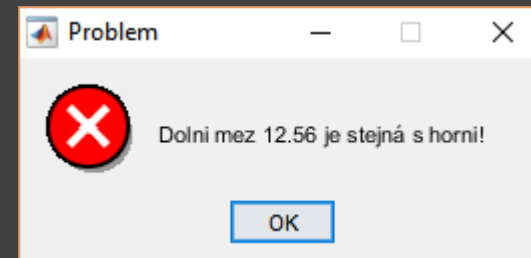
a už se vykreslí graf
s příslušným titulkem ...



nebo:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: $4 * \pi - 0.01$



Funkce pro vstup a výstup

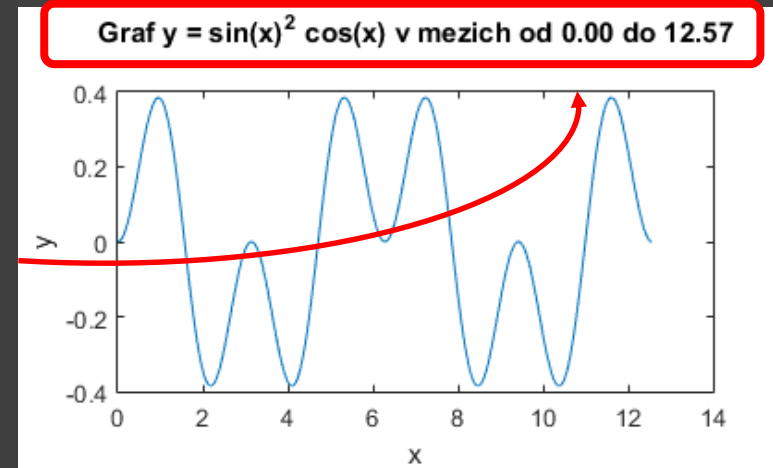
Pokračování příkladu:

Volání funkce jejím názvem:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **0**

a už se vykreslí graf
s příslušným titulkem ...

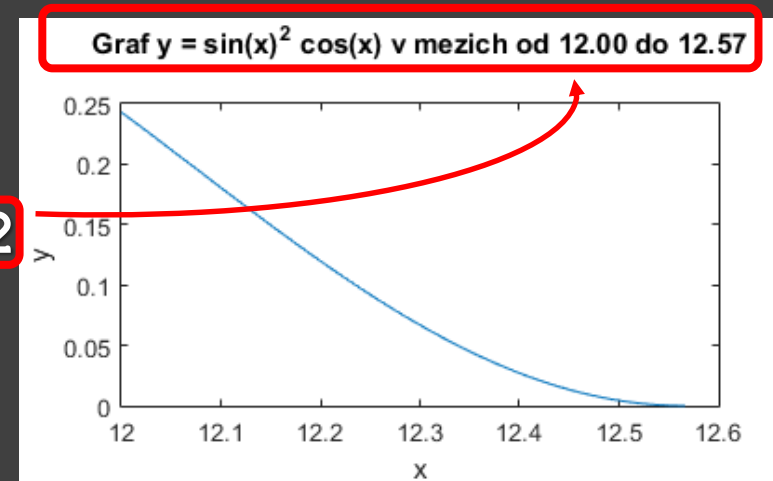


nebo:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **12**

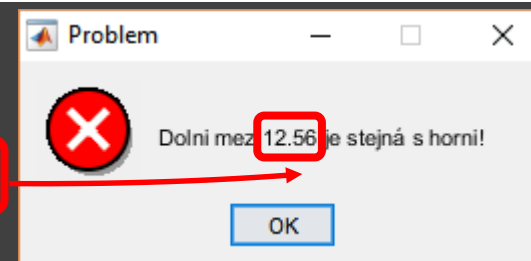
a už se vykreslí graf
s příslušným titulkem ...



nebo:

`s2cos_graf_s_titulkem`

Zadejte počáteční hodnotu: **$4 * \pi - 0.01$**



Funkce pro vstup a výstup

Příklad:

Funkce – jednoduchá kalkulačka

Uživatel zadá dvě čísla a bude vybírat z operací:

$+$, $-$, $*$, $/$, \backslash .

Pokud uživatel zadá matici nebo vektor, jako zadané číslo je uvažován prvek v prvním řádku a prvním sloupci.

Uživateli bude umožněno po skočení výpočtu provádět další výpočet nebo zadat jiná čísla a opět pokračovat ve výpočtech.

```

function kalkulacka
while(1)
    a = input('Zadej prvni cislo:'); if (length(a)>1) a = a(1,1); end
    b = input('Zadej druhe cislo:'); if (length(b)>1) b = b(1,1); end
    while(1)
        znak = menu('Vyber', '+', '-', '*', '/', '\', 'Jina cisla', 'Konec');
        switch (znak)
            case 1
                s = sprintf('%0.1f + %0.1f = %0.2f', a, b, a+b);
            case 2
                s = sprintf('%0.1f - %0.1f = %0.2f', a, b, a-b);
            case 3
                s = sprintf('%0.1f * %0.1f = %0.2f', a, b, a.*b);
            case 4
                s = sprintf('%0.1f / %0.1f = %0.2f', a, b, a./b);
            case 5
                s = sprintf('%0.1f \ \ %0.1f = %0.2f', a, b, a.\b);
            case 6
                msgbox('Zadej jina cisla', 'Informace', 'warn'); break;
            case 7
                msgbox('Konec', 'Informace', 'warn'); return;
        end
        msgbox(s, 'Vypocet', 'help');
    end
end
end
end

```

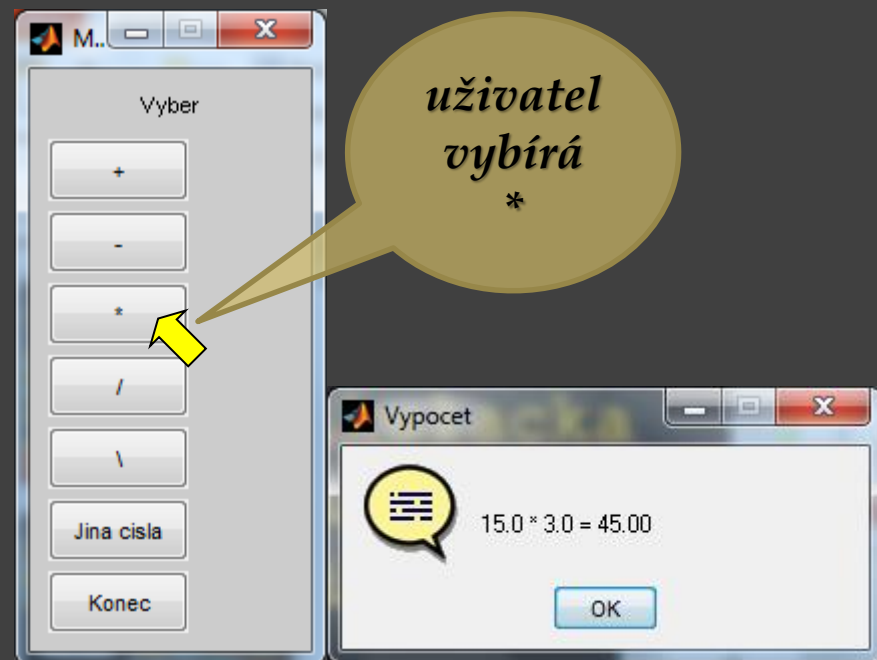
Funkce pro vstup a výstup

Pokračování příkladu:

Volání funkce **kalkulacka**

Zadej první číslo:15

Zadej druhé číslo:3



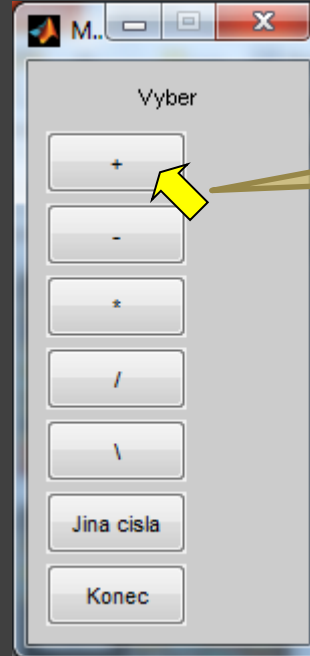
Funkce pro vstup a výstup

Pokračování příkladu:

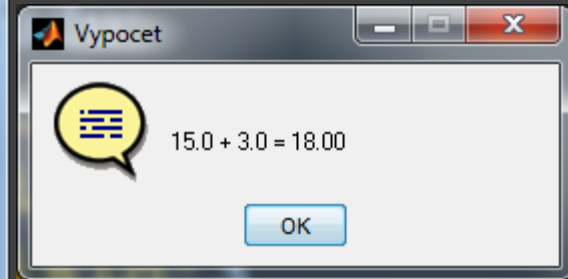
Volání funkce **kalkulacka**

Zadej první číslo: 15

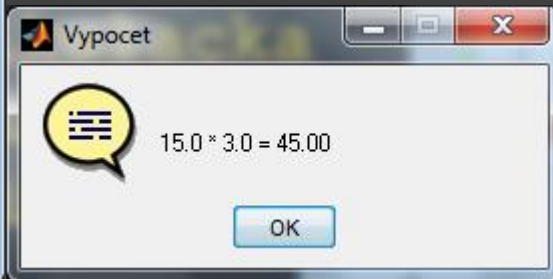
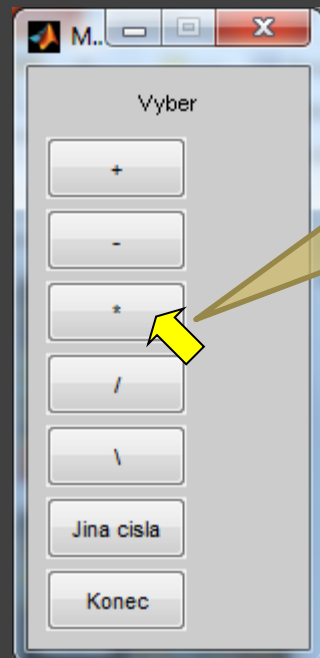
Zadej druhé číslo: 3



*uživatel
vybírá
+*



*uživatel
vybírá
**



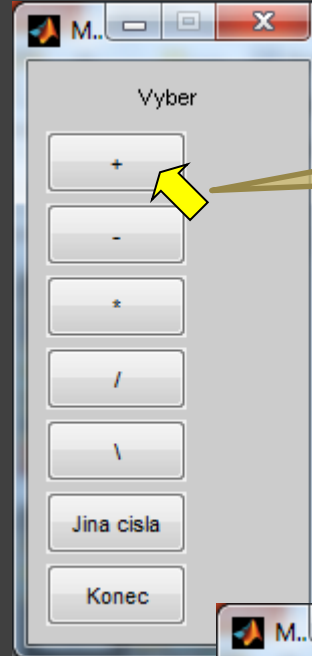
Funkce pro vstup a výstup

Pokračování příkladu:

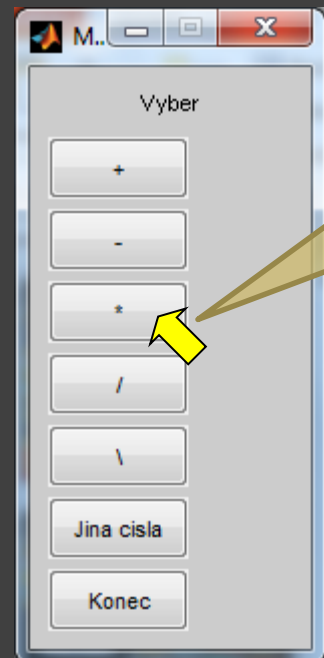
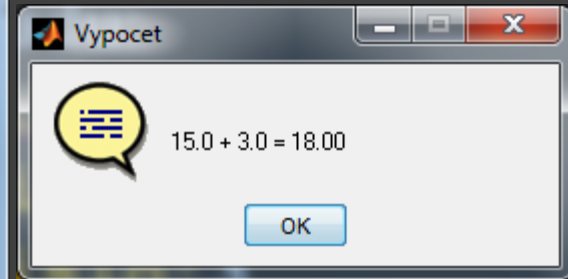
Volání funkce **kalkulacka**

Zadej první číslo: 15

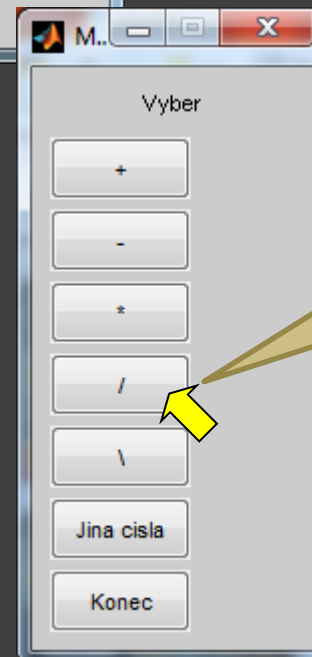
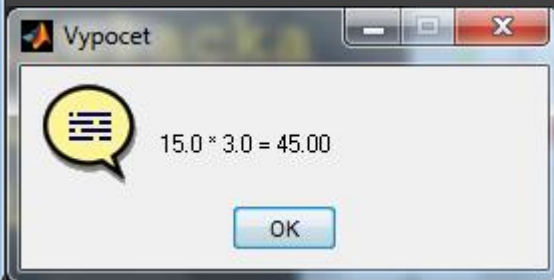
Zadej druhé číslo: 3



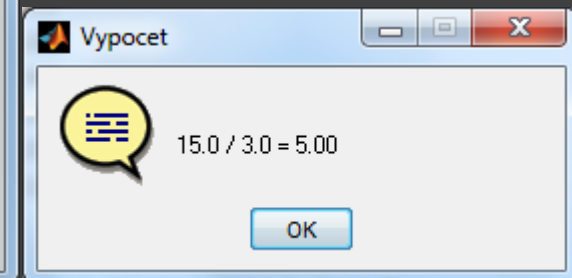
*uživatel
vybírá
+*



*uživatel
vybírá
**



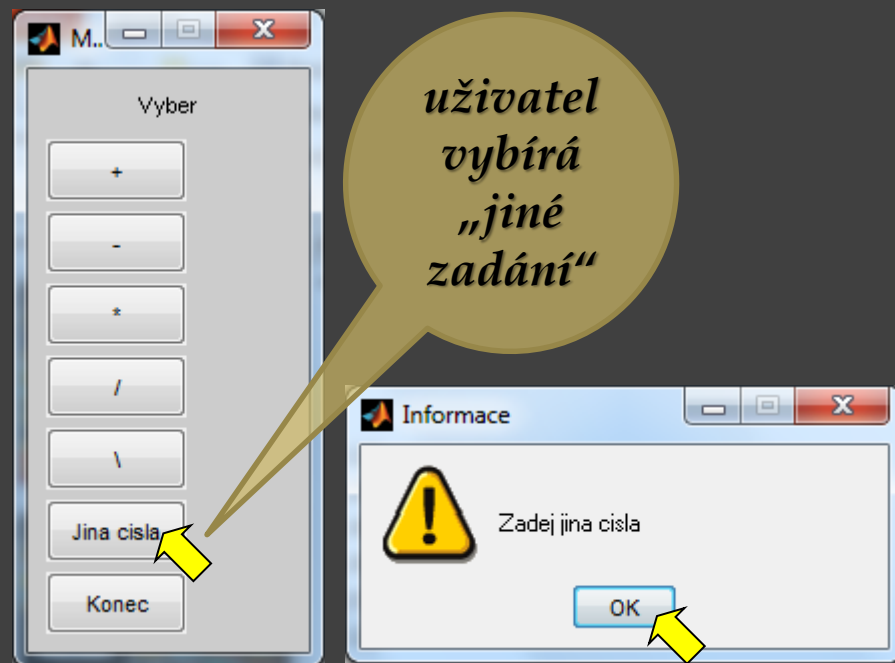
*uživatel
vybírá
/*



Funkce pro vstup a výstup

Pokračování příkladu:

Volání funkce **kalkulacka**

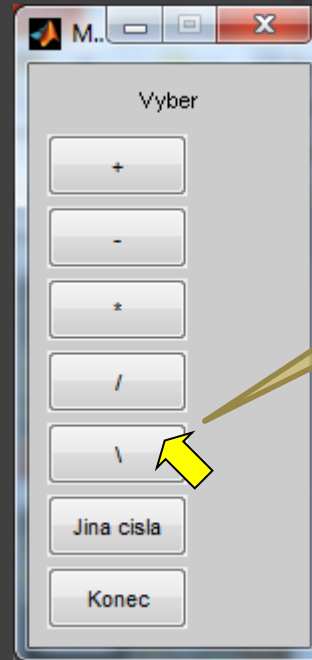


Zadej první číslo:1

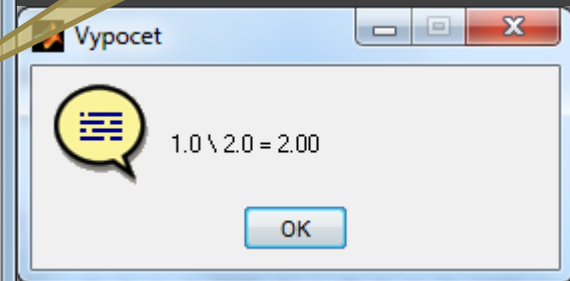
Zadej druhé číslo:2

Funkce pro vstup a výstup

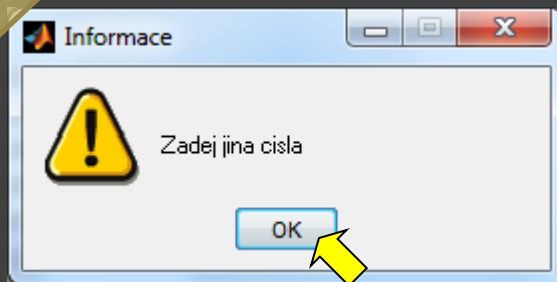
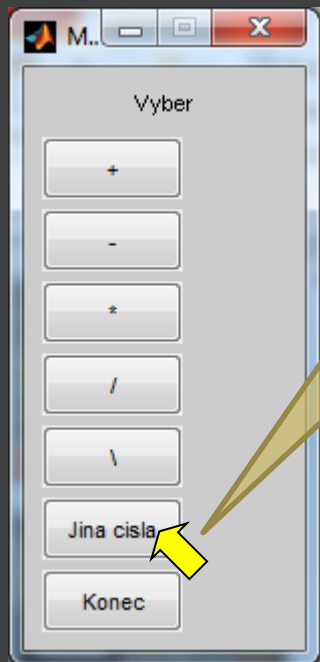
Pokračování příkladu:
Volání funkce **kalkulacka**



*uživatel
vybírá
\
/*



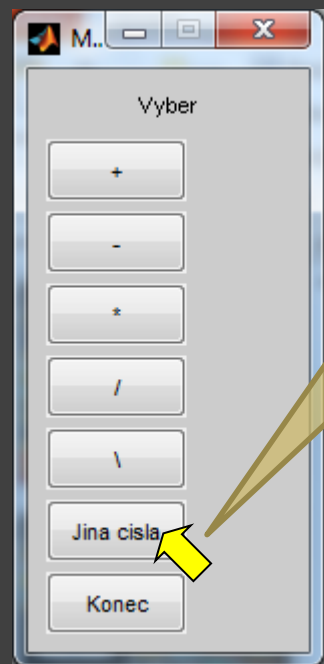
*uživatel
vybírá
„jiné
zadání“*



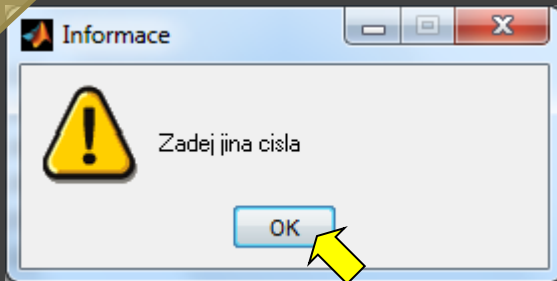
Zadej první číslo:1
Zadej druhé číslo:2

Funkce pro vstup a výstup

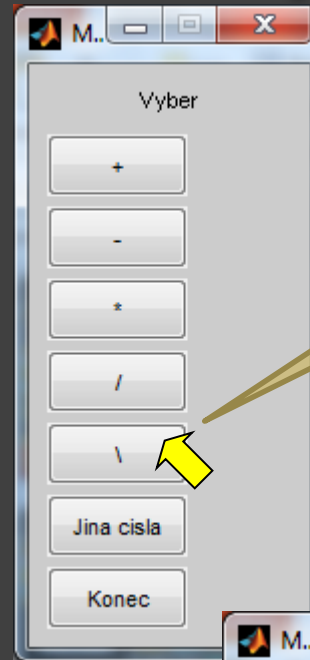
Pokračování příkladu:
Volání funkce **kalkulacka**



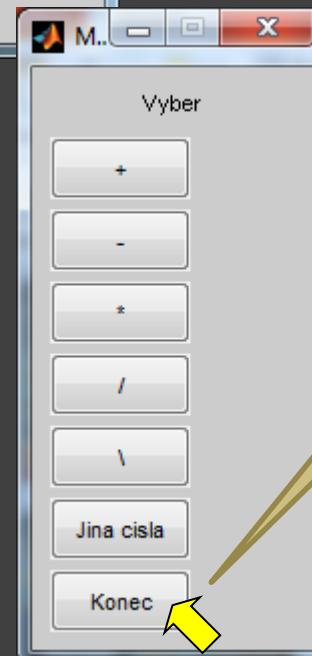
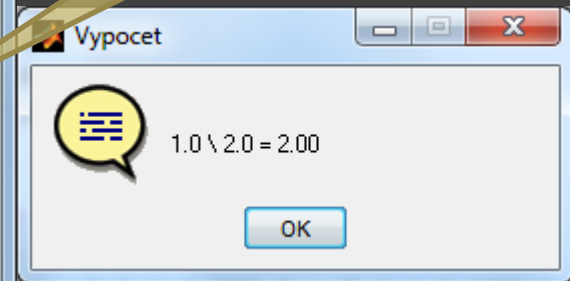
*uživatel
vybírá
„jiné
zadání“*



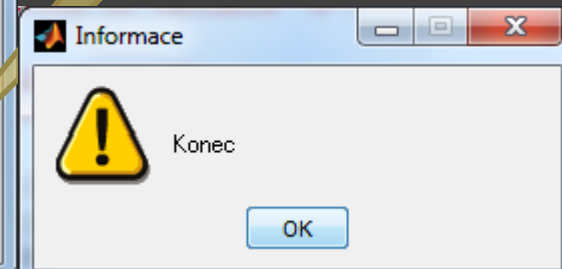
Zadej první číslo:1
Zadej druhé číslo:2



*uživatel
vybírá
*



*uživatel
vybírá
„Konec“*



Základy práce se soubory

```
f = fopen('nazev_soubor', 'režim');
```

fopen otevře soubor s názvem `nazev_soubor` pro přístup pro čtení nebo zápis podle zvoleného režimu. Řetězec `nazev_soubor` obsahuje název souboru, který má být otevřen.

Pokud `fopen` nemůže otevřít soubor, vrátí hodnotu `-1`. Jetliže soubor není v aktuální složce, lze zadat i cestu k souboru, např.:

```
f = fopen('C:\\cesta_k_souboru\\soubor', 'režim');
```

kde **režim** může být:

'r' – číst (**r**ead)

'w' – zápis, přepis, vytvoření nového souboru (**w**rite)

'a' – připsat na konec existujícího (**a**ppend)

'r+' – čtení nebo zápis

'w+' – čtení nebo zápis, přepis, vytvoření nového souboru

'a+' – čtení nebo zápis, přepis, vytvoření nového souboru a přidávat na konec souboru další data