

Vysoká škola báňská – Technická univerzita Ostrava
Fakulta stavební



Přednáška z předmětu: Algoritmizace inženýrských výpočtů

Téma č.1: Matlab

prof. Ing. Martin Krejsa, Ph.D.

Obsah

1. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

Obsah

1	Matlab	3
1.1	Zadání proměnných	6
1.2	Vektory a matice	6
1.2.1	Přístup k maticím a vektorům	8
1.2.2	Maticové operace	8
1.3	Správa proměnných	9
1.4	Využití grafického výstupu	9
1.4.1	Graf funkce	10
1.5	Vytvoření skriptů	11
1.5.1	Příkazy cyklu	12
1.5.2	Logické rozhodování	12
	Literatura	23



Obsah

2. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno



Kapitola 1

Matlab

Cíle

Kapitola je zaměřena na:

- seznámení s uživatelským prostředím systému MATLAB,
- definici a správu proměnných,
- úvodní ukázkou tvorby algoritmu.

MATLAB [3] (viz obr. 1.1) je programové prostředí využívající skriptovací programovací jazyk pro vědeckotechnické numerické výpočty, modelování a návrhy algoritmů. Nástavbou systému MATLAB je Simulink — program pro simulaci a modelování dynamických systémů, který využívá algoritmy Matlabu pro numerické řešení především nelineárních diferenciálních rovnic.

Název MATLAB vznikl ze zkratky MATRIX LABORATORY (volně přeloženo jako „maticová

Obsah

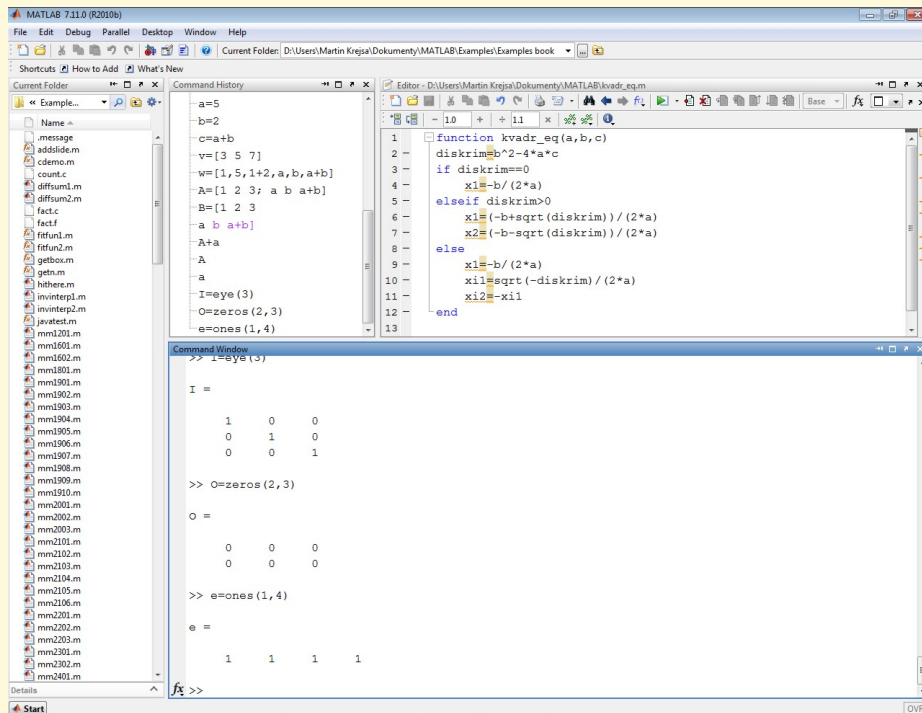
3. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno



Obr. 1.1 Pracovní plocha programu MATLAB

laboratoř“), což vychází ze skutečnosti, že klíčovou datovou strukturou při výpočtech v systému MATLAB jsou matice.



Obsah

4. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

Pro zájemce

MATLAB je komerční software, takže vítanou alternativou pro tvorbu algoritmů s využitím kompatibilních příkazů a skriptů je software nazývaný Octave, který je zdarma ke stažení např. z [1]. Program Octave vznikl jako open-source a existuje tedy v mnoha mutacích pro platformy Unix (Linux), Mac OS nebo MS Windows. Uživatelsky poněkud strohé prostředí programu obsahuje pouze příkazový řádek v textovém režimu pro zadání konkrétního příkazu a výpis výsledných hodnot.

Samotný systém MATLAB obsahuje řadu příkazů pro správu proměnných, základní operace algebry, výpočty s vektory i maticemi, a příkazy vyšší matematiky. Vzhledem k poněkud rozsáhlým možnostem je k dispozici rozsáhlá nápověda, se kterou lze pracovat s pomocí příkazů obsažených v tab. 1.1.

<i>Název proměnné</i>	<i>Popis</i>
<code>help</code>	vyvolání obsahu nápovědy
<code>help příkaz</code>	vyvolání nápovědy, vztahující se ke konkrétnímu příkazu
<code>lookfor výraz</code>	výpis všech položek nápovědy, které obsahují hledaný výraz
<code>info</code>	informace o firmě MathWorks

Tab. 1.1 Přehled příkazů souvisejících s nápovědou

MATLAB si uchovává přesnou reprezentaci veškerých číselných hodnot, se kterými pracuje. Na uživateli však záleží, v jakém formátu jsou tyto hodnoty zobrazovány, k čemuž využívá příkaz `format` s příslušnou specifikací podle tab. 1.2.



Obsah

5. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

1.1. Zadání proměnných

Základním typem proměnné v systému MATLAB je matice. Jednoduchá proměnná, obsahující jednu hodnotu, je interpretována jako matice typu $[1, 1]$. Proměnné se zadávají příkazy, které se definují v příkazovém řádku programu. Končí-li příkaz středníkem, výsledek příkazu se nezobrazí. Pokud je součástí zadání základní aritmetická operace, lze pro ně využít symboly z tab. 1.3. Desetinnou čárku lze zadat pomocí tečky.

Definice proměnné probíhá automaticky přiřazením její hodnoty pomocí rovnítka, např.:

```
a=5
b=2
c=a+b
d='Výsledek'
```

Pro přiřazení hodnoty proměnným lze využít i předem nadefinovaných speciálních proměnných z tab. 1.4

K dispozici je rovněž množství elementárních funkcí s jedním vstupním parametrem. Přehled nejzákladnějších je uveden v tab. 1.5.

1.2. Vektory a matice

K definici vektoru se používají hranaté závorky. Mezery nebo čárky oddělují prvky v řádku, např.:

```
u=[3 5 7]
v=[1,5,1+2,a,b,a+b]
w=[0,pi,2*pi]
```



Obsah

6. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

K vytvoření vektorů lze využít i pokročilejších technik, popsanych v tab. 1.6

Definice matice se provádí podobným způsobem, jak je tomu u vektorů. Řádky se ale oddělují pomocí středníku nebo klávesou <Enter>, např.:

```
A=[1 2 3; a b a+b]
```

```
B=[1 2 3
```

```
a b a+b]
```

Poznámka 1.1. MATLAB rozlišuje malá a velká písmena v názvech proměnných. Příkaz

```
A+a
```

vygeneruje pro předchozí zadání matice **A** a proměnné **a**:

```
ans =
```

```
6      7      8  
10     7     12
```

tzn., že ke každému prvku matice **A** se přičte obsah proměnné **a**, tedy hodnota 5.

K přímému generování matic nebo vektoru s předepsaným rozměrem lze využít některé ze standardních funkcí systému MATLAB, např.:

```
I=eye(3)
```

```
0=zeros(2,3)
```

```
e=ones(1,4)
```

V prvním případě je vygenerována čtvercová matice s hodnotami 1 na diagonále a 0 mimo diagonálu. Následuje vytvoření matice, resp. vektoru s hodnotou 0 resp. 1 ve všech jejích prvcích.



Obsah

7. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

1.2.1. Přístup k maticím a vektorům

Po vytvoření vektoru nebo matice se lze k jednotlivým prvkům odkazovat. V případě předchozí matice $A=[1 \ 2 \ 3; a \ b \ a+b]$ (konkrétní obsah je tedy $[1 \ 2 \ 3; 5 \ 2 \ 7]$) jsou na ukázkou možné variace odkazů popsány v tab. 1.7.

1.2.2. Maticové operace

S vektory i maticemi lze provádět maticové operace. Transponování matic se provádí s využitím operátoru ' (apostrof), např. příkaz A' transponuje původní matici A :

```
ans =  
    1    5  
    2    2  
    3    7
```

K provedení operací mezi jednotlivými prvky matice nebo vektoru se musí umístit znak . (tečka) před operátor. Například pro dříve zadaný vektor $u=[3 \ 5 \ 7]$ je výsledkem příkazu $u*u'$:

```
ans = 83
```

kdežto po zadání příkazu $u.*u$ se zobrazí vektor:

```
ans =  
    9    25    49
```



Obsah

8. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

Podobně lze provádět i další operace s vektory a maticemi prvek po prvku, jak je pro obecně definované vektory $a = a_1, a_2, \dots, a_n$; $b = b_1, b_2, \dots, b_n$ a skalár c uvedeno v tab. 1.8.

K dispozici je také několik maticových funkcí. Výčet nejzákladnějších obsahuje tab. 1.9.

1.3. Správa proměnných

Pro správu zadaných proměnných, vektorů a matic je v prostředí systému MATLAB zavedeno několik příkazů:

- příkazy `who` nebo `whos` vypíše seznam všech aktuálně definovaných proměnných (druhý z nich i s jejich velikostmi),
- příkaz `size`(proměnná) vrátí rozměry dané proměnné,
- příkaz `clear`(proměnná) vymaže danou proměnnou z paměti,
- příkaz `clear` bez parametru vymaže všechny zadané proměnné z paměti.

1.4. Využití grafického výstupu

Základním nástrojem v grafickém režimu je příkaz `plot` se syntaxí `plot(x,y,options)`, kde x a y jsou souřadnice bodů, které se mají vykreslit. Parametr `options` definuje způsob, jakým se bude grafický výstup provádět. Obsahuje znaky pro definici barvy a stylu vykreslení bodů, případně jejich spojnice:



Obsah

9. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

- barva bodů a jejich spojnice: **b** (modrá), **g** (zelená), **r** (červená), **c** (modrozelená), **m** (fialová), **y** (žlutá), **k** (černá), **w** (bílá);
- styl spojnice bodů: **-** (body budou spojeny plnou čarou), **:** (body budou spojeny tečkovanou čarou), **-.** (body budou spojeny čerchovanou čarou), **--** (body budou spojeny přerušovanou čarou);
- styl vykreslení bodů: ***** (body budou vykresleny jako hvězdičky), **.** (tečky), **x** (křížky), **+** (znaky plus), **o** (kolečka), **s** (čtverce), **d** (kosočtverce).

Příklad 1.2. Pro vykreslení spojnice pěti bodů o souřadnicích z tab. 1.10 poslouží příkaz:
`plot([10:15],[1,5,7,4,6,10], 'r-*')`

Výsledný grafický výstup je zobrazen na obr. 1.2.

V souvislosti s grafickým prostředím je užitečný i příkaz `hold on`, který umožňuje grafický výstup více příkazů do jednoho okna (do původního stavu pak vše vrátí příkaz `hold off`).

1.4.1. Graf funkce

Při vytváření grafu funkce je nutné nejprve diskretizovat osu x . V získaných bodech je pak nutno určit odpovídající funkční hodnoty $f(x)$. Graf diskretizované funkce se pak zobrazí již známým příkazem `plot`.

Příklad 1.3. Vykreslení grafu funkce sinus lze s využitím příkazu `plot` provést následující sekvencí příkazů:



Obsah

10. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

```
x=linspace(0,2*pi,30);  
y=sin(x);  
plot(x,y,'b-')
```

Graf lze doplnit o nadpis, případně popisem os, následujícím doplněním:

```
title('Graf funkce y=sin(x)');  
xlabel('x');  
ylabel('sin(x)')
```

Výsledné zobrazení grafu funkce sinus je na obr. 1.3.

1.5. Vytvoření skriptů

V programu MATLAB lze posloupnost příkazů zadávat jednoduše do příkazového řádku, systém interaktivně tyto povely postupně zpracovává. Pokud by se měl ale výpočet provádět opakovaně, musí se veškeré příkazy zadávat znovu, což je pracné a nevhodné.

Řešením je uložení sledu matlabovských příkazů, tzv. **skriptů** do textového souboru s příponou ***.m** (tzv. m-soubor). S využitím příkazů, uložených v m-souboru, se výpočet spustí zadáním názvu souboru (bez přípony), přičemž se prohledává aktuální adresář a adresáře uvedené v seznamu systémové proměnné **path**.

Tímto způsobem lze vytvořit v samostatném souboru i speciální výpočetní funkci (tzv. „m-funkci“), kterou lze vyvolat z příkazového řádku zadáním názvu funkce, případně seznamem vstupních parametrů v závorce, oddělených čárkami. **Název souboru by měl být shodný s názvem funkce v jeho záhlaví.**

M-soubory mohou obsahovat i tzv. řídicí příkazy, typické pro pokročilejší programovací platformy. Patří k nim zejména příkazy cyklu a logického rozhodování.



Obsah

11. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

1.5.1. Příkazy cyklu

MATLAB umožňuje použití dvou typů programových cyklů:

- **for cyklus**, jehož syntaxe je

```
for i=počáteční hodnota:krok:koncová hodnota
    posloupnost příkazů
end
```

- **while cyklus** se syntaxí:

```
while logická podmínka
    posloupnost příkazů
end
```

Podrobnější výklad tvorby algoritmů s využitím obou typů cyklů je obsažen v následujících kapitolách.

1.5.2. Logické rozhodování

Syntaxe posloupnosti příkazů, které mají být rozděleny do tří bloků podle výsledků logického rozhodování, je následující:

```
if logická podmínka 1
    posloupnost příkazů 1
```



Obsah

12. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

```
elseif logická podmínka 2
    posloupnost příkazů 2
else
    posloupnost příkazů 3
end
```

Při definování logických podmínek, jejichž výsledkem je buď „pravda“ nebo „nepravda“, je možno využít následující logické operátory: & (and), | (or), ~ (not). Relačními operátory mohou být: <, <=, >=, > a == (rovnost).

Příklad 1.4. K procvičení tvorby m-funkce poslouží řešení kvadratické rovnice, kterou lze s využitím logických operátorů vytvořit následujícím způsobem:

```
function kvadr_eq(a,b,c)
diskrim=b^2-4*a*c
if diskrim==0
    x1=-b/(2*a)
elseif diskrim>0
    x1=(-b+sqrt(diskrim))/(2*a)
    x2=(-b-sqrt(diskrim))/(2*a)
else
    x1=-b/(2*a)
    xi1=sqrt(-diskrim)/(2*a)
    xi2=-xi1
end
```

Funkce, kterou lze v příkazovém řádku programu MATLAB vyvolat se vstupními parametry zadáním např. `kvadr_eq(5,10,1)`, pak vrátí dva reálné kořeny kvadratické rovnice:

[Obsah](#)[13. strana ze 23](#)[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)

```
diskrim =  
      80
```

```
x1 =  
-0.105572809000084
```

```
x2 =  
-1.894427190999916
```

Při vyvolání vytvořené funkce s parametry `kvadr_eq(10,5,1)` lze naopak získat výsledek:

```
diskrim =  
     -15
```

```
x1 =  
-0.250000000000000
```

```
xi1 =  
 0.193649167310371
```

```
xi2 =  
-0.193649167310371
```

Dosažené výsledky lze zkontrolovat speciální funkcí `roots(c)`, kde `c` je vektor koeficientů polynomu seřazených sestupně podle mocnin x . Při obecném vyjádření polynomu n -tého stupně:

$$f(x) = c_n \cdot x^n + c_{n-1} \cdot x^{n-1} + \dots + c_1 \cdot x + c_0, \quad (1.1)$$

[Obsah](#)[14. strana ze 23](#)[Zavřít dokument](#)[Konec](#)[Celá obrazovka/Okno](#)

obsahuje vektor c prvky $c_n, c_{n-1}, \dots, c_1, c_0$.

Pro první výpočet příkladu 1.4 je pak sled příkazů následující:

```
c=[5,10,1]
roots(c)
```

s výsledkem

```
ans =
-1.894427190999916
-0.105572809000084
```

Poznámka 1.5. Pro popis práce s programovým systémem MATLAB, resp. Octave, existuje množství publikací. Některé z nich jsou přeloženy do češtiny a v elektronické podobě jsou volně přístupné (např. [2, 4]).



Obsah

15. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

<i>Příkaz</i>	<i>Popis formátu</i>	<i>Ukázka zobrazení čísla π</i>
format short	pevná desetinná čárka, 5 zobrazených číslic	3.1416
format long	pevná desetinná čárka, 15 zobrazených číslic (typ proměnné double), resp. 7 (typ single)	3.141592653589793
format shorte	pohyblivá desetinná čárka, 5 zobrazených číslic	3.1416e+000
format longe	pohyblivá desetinná čárka, 15 zobrazených číslic (typ proměnné double), resp. 7 (typ single)	3.141592653589793e+000
format shorteng	inženýrský formát, 5 zobrazených číslic a exponent	3.1416e+000
format longeng	inženýrský formát, 16 zobrazených číslic a exponent	3.14159265358979e+000
format rat	vyjádření výsledku formou zlomku (implicitní nastavení)	355/113
format hex	zobrazení v šestnáctkové soustavě	400921fb54442d18

Tab. 1.2 Přehled možných typů formátů zobrazení číselných hodnot



Obsah

16. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

<i>Operace</i>	<i>Symbol</i>	<i>Příklad</i>
Součet	+	4+11, a+b
Rozdíl	-	18-5, a-b
Součin	*	7.13*5, a*b
Podíl ($\frac{1}{8}$)	/ nebo \	1/8 = 8\1
Mocnina (2^8)	^	2^8

Tab. 1.3 Přehled symbolů základních aritmetických operací

<i>Název proměnné</i>	<i>Popis</i>
ans	proměnná, která obsahuje výsledek aritmetické operace
pi	Ludolfovo číslo π
inf	označení pro nekonečno ∞
nargin	počet vstupních parametrů dané funkce
nargout	počet výstupních parametrů dané funkce
eps	nejmenší použitelné číslo v daném formátu
realmin	absolutně nejmenší použitelné kladné reálné číslo
realmax	absolutně největší použitelné kladné reálné číslo

Tab. 1.4 Přehled předem nadefinovaných proměnných



Obsah

17. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno

<i>Název funkce</i>	<i>Popis</i>
<code>abs(x)</code>	absolutní hodnota z x
<code>cos(x)</code> , <code>sin(x)</code> , <code>tan(x)</code>	goniometrické funkce (parametr x v radiánech)
<code>acos(x)</code> , <code>asin(x)</code> , <code>atan(x)</code>	inverzní goniometrické funkce
<code>exp(x)</code>	exponenciální funkce (e^x)
<code>log(x)</code>	přirozený logaritmus x
<code>log10(x)</code>	dekadický logaritmus x
<code>sqrt(x)</code>	druhá odmocnina x

Tab. 1.5 Přehled základních elementárních funkcí

<i>Příkaz</i>	<i>Popis</i>	<i>Výsledek</i>
<code>x=[1:5]</code>	vytvoří řádkový vektor x , začínající hodnotou 1, končící hodnotou 5, přičítá se hodnota 1	<code>x=[1,2,3,4,5]</code>
<code>x=[2:3:11]</code>	vytvoří řádkový vektor x , začínající hodnotou 2, končící hodnotou 11, přičítá se hodnota 3	<code>x=[2,5,8,11]</code>
<code>x=linspace(1,25,5)</code>	vytvoří řádkový vektor x , začínající hodnotou 1, končící hodnotou 25, vektor obsahuje 5 prvků	<code>x=[1,7,13,19,25]</code>

Tab. 1.6 Příkazy pro konstrukci vektorů



Obsah

18. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno



<i>Příkaz</i>	<i>Popis</i>	<i>Výsledek</i>
<code>A(2,1)</code>	prvek z 2. řádku a 1. sloupce matice A	5
<code>A(2,[1 3])</code>	1. a 3. prvek z 2. řádku matice A	5, 7
<code>A(1,1:3)</code>	prvky 1 až 3 na 1. řádku matice A	1, 2, 3
<code>A(1,1:end)</code>	stejně jako předchozí	1, 2, 3
<code>A(1,:)</code>	stejně jako předchozí	1, 2, 3
<code>A(2,1:2:3)</code>	1. a 3. prvek z 2. řádku matice A, <code>a:b:c</code> definuje aritmetickou posloupnost s prvním prvkem rovným <code>a</code> , posledním rovným <code>c</code> , <code>b</code> je diference mezi sousedními prvky (pokud je rovno 1, nemusí se uvádět)	5, 7
<code>A(1:end,2:end)</code>	2. a 3. prvek z obou řádků (submatice)	2, 3; 2, 7

Tab. 1.7 Příkazy pro přístup k prvkům vektorů a matic

Obsah

19. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno



<i>Operace</i>	<i>Příkaz</i>	<i>Výsledek</i>
skalární součet	$\mathbf{a+c}$	$a_1 + c, a_2 + c, \dots, a_n + c$
skalární součin	$\mathbf{a*c=a.*c}$	$a_1 \cdot c, a_2 \cdot c, \dots, a_n \cdot c$
vektorový součet	$\mathbf{a+b}$	$a_1 + b_1, a_2 + b_2, \dots, a_n + b_n$
vektorový součin	$\mathbf{a.*b}$	$a_1 \cdot b_1, a_2 \cdot b_2, \dots, a_n \cdot b_n$
vektorové dělení zleva	$\mathbf{a./b}$	$a_1/b_1, a_2/b_2, \dots, a_n/b_n$
vektorové dělení zprava	$\mathbf{a.\backslash b}$	$a_1 \backslash b_1, a_2 \backslash b_2, \dots, a_n \backslash b_n$
skalární mocnění	$\mathbf{a.^c}$	$a_1^c, a_2^c, \dots, a_n^c$
skalární mocnění	$\mathbf{c.^a}$	$c^{a_1}, c^{a_2}, \dots, c^{a_n}$
vektorové mocnění	$\mathbf{a.^b}$	$a_1^{b_1}, a_2^{b_2}, \dots, a_n^{b_n}$

Tab. 1.8 Operace s vektory a maticemi prvek po prvku

<i>Název funkce</i>	<i>Popis</i>
$\mathbf{det(A)}$	determinant matice A
$\mathbf{inv(A)}$	výpočet inverzní matice k A
$\mathbf{diag(A)}$	výpis prvků na diagonále matice A

Tab. 1.9 Přehled základních maticových funkcí

x	10	11	12	13	14	15
y	1	5	7	4	6	10

Tab. 1.10 Vstupní hodnoty souřadnic pro vykreslení spojnice bodů

Obsah

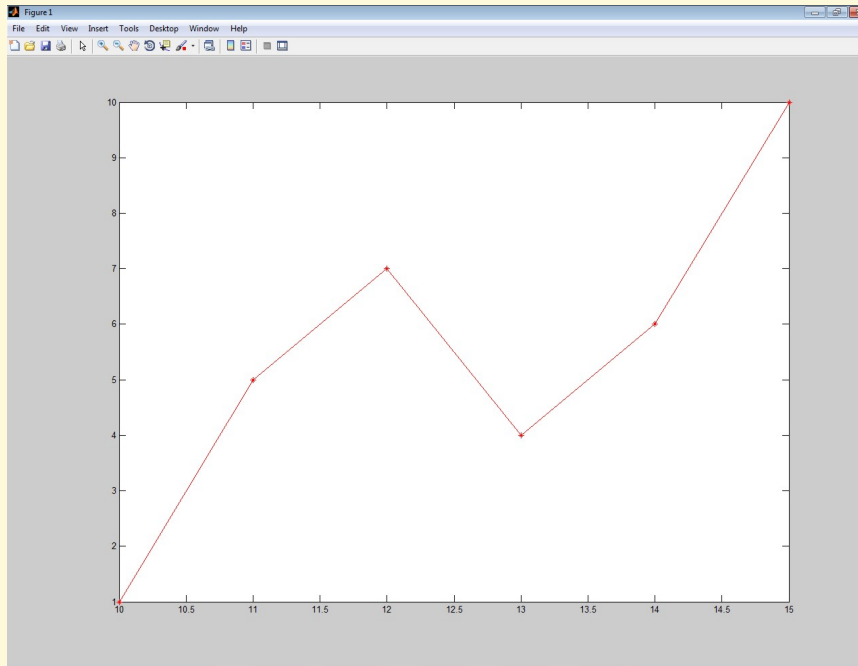
20. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno



Obr. 1.2 Ukázka grafického výstupu z programu MATLAB



Obsah

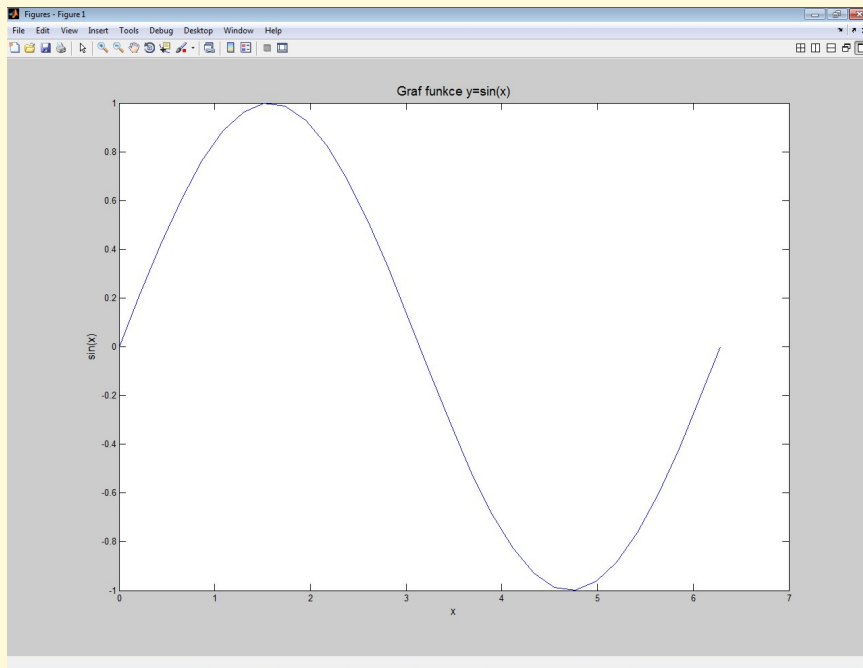
21. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno



Obr. 1.3 Graf funkce sinus



Obsah

22. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno



Literatura

- [1] Eaton, J.W. *Octave*. Programový systém pro provádění matematických výpočtů. Freeware, verze 4.2.1. [on-line]. <<http://www.gnu.org/software/octave>>. University of Wisconsin, Department of Chemical Engineering, 1998-2017. (Citováno na s 5.)
- [2] Just, M. *Octave — český průvodce programem*. Elektronický manuál programového systému Octave. [on-line]. <<http://www.octave.cz>>. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky, 2006. (Citováno na s 15.)
- [3] MATLAB. Programový systém pro provádění matematických výpočtů. Komerční software, verze R2017b. [on-line]. <<http://www.mathworks.com>>. The MathWorks, prosinec 2017. (Citováno na s 3.)
- [4] Sigmon K. *MATLAB Primer CZ*. Elektronický manuál programového systému MATLAB. Druhé vydání. [on-line]. <<https://artax.karlin.mff.cuni.cz/~beda/cz/matlab/primercz/matlab-primer.html>>. Department of Mathematics, University of Florida, 1989, 1992. Z anglického originálu přeložil Petr Klášterecký. (Citováno na s 15.)

Obsah

23. strana ze 23



Zavřít dokument

Konec

Celá obrazovka/Okno