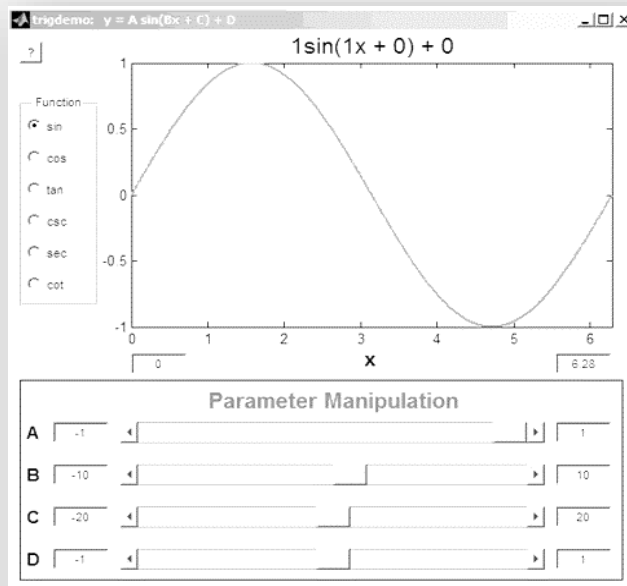


Počítačové aplikácie

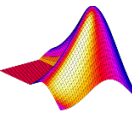
Pr.7 – Tvorba skriptov, programovanie



1. roč. AES, LS 2018/19

Náplň

1. Tvorba m-súborov
2. Typy m-súborov
3. Riadiace príkazy – príkaz podmienky
 - cyklu a jeho prerušenia
 - chybového hlásenia
 - pre vstup z klávesnice
 - pre výstup na obrazovku
 - prerušenia výpočtu
4. Príkazy pre používateľský vstup
 - input
 - keyboard
 - pause



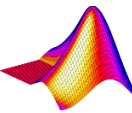
1. Tvorba m-súborov

M-súbory sú:

- súbory, ktoré sú vytvorené zabudovaným editorom MATLABu (M-editor/Debugger) a **majú príponu *.m**
- m-súbory obsahujú zoznam užívateľom zapísaných príkazov a povelov systému MATLAB (využívajú jednotlivé inštrukcie)

Dôvody tvorby m-súborov:

- **ukladanie postupností príkazov** (skript) na disk alebo ukladanie **užívateľských funkcií**
je tu možnosť spustiť celý súbor z prostredia MATLABu
- **spustenie** jedného, či viac súborov z iného m-súboru
- ďalšie možnosti (**programovanie, tvorba grafických objektov** a pod.)

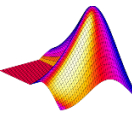


2. Typy m-súborov

- skripty

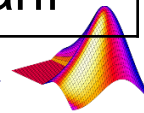
- funkcie

Vlastnosti skriptu	Vlastnosti funkcie
Nezadávajú sa vstupné argumenty a žiadne výstupné hodnoty sa nevypisujú	Môže obsahovať vstupné a vracať výstupné premenné
Pracuje s premennými vo Workspace – vytvára nové, – maže a mení existujúce, Výsledky sú k dispozícii po jeho skončení.	Vnútorne premenné sú lokálne pre funkciu
Vhodné pre automatizáciu volania tých istých príkazov viackrát	Vhodné pre rozšírenie funkčnej základne MATLABovských aplikácií



3. Riadiace príkazy – prehľad

Príkaz	Opis
break	ukončenie vykonávaného cyklu
else	súvisiace s if
elseif	súvisiace s if
end	ukončenie rozsahu príkazov for , while , switch a if
for	opakované vykonávanie príkazov so špecifikovaným krokom
error	zobrazenie správy a ukončenie cyklu for alebo while
if	podmienené vykonávanie príkazov
switch	podmienené vetvenie príkazov
return	návrat do volajúcej funkcie
while	podmienené opakovanie vykonávania príkazov
warning	zobrazenie chybovej hlášky, program pokračuje vo vykonávaní



Riadiace príkazy - operátory

Používanie operátorov v porovnávacích výrazoch:

Matematický operátor	Popis	Operátor v MATLABe
<	menší než	<
≤	menší než alebo rovný	<=
>	väčší než	>
≥	väčší než alebo rovný	>=
=	rovný	==
≠	nerovný	~=

Pozor pri porovnávaní matic!

nepoužívať: if A == B

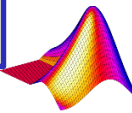
ale používať: if isequal(A, B)

Vyskúšajte:

```
A = zeros(3,3)+1e-20;
```

```
B = zeros(3,3);
```

```
Vysl = isequal(A,B)
```



Riadiace príkazy – if, else, elseif

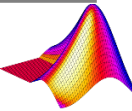
testuje splnenie logickej podmienky/podmienok.

Syntax:

```
if logická podmienka  
    príkazy (ak je splnená  
        podmienka)  
end
```

```
if logická podmienka  
    príkazy (ak je splnená  
        podmienka)  
else  
    príkazy (ak nie je splnená  
        podmienka)  
end
```

```
if logická podmienka1  
    príkazy (ak je splnená  
        podmienka1)  
elseif logická podmienka2  
    príkazy (ak je splnená  
        podmienka2)  
else  
    príkazy (ak nie je splnená  
        podmienka1  
        ani podmienka2)  
end
```



Riadiace príkazy – príklad na if, else, elseif

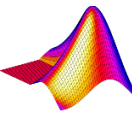
Zadanie: vygenerujte náhodné číslo z intervalu $<-10, 10>$.
Ak číslo kladné, potom vypíše informáciu, či je párne alebo nepárne. Ak je nepárne, vypíšte iba že je nepárne.

Poznámka: inštrukcia `rem = zvyšok` po delení (`remainder = zvyšok`).

```
A = round(rand(1)*20 - 10)
if rem (A,2) == 0
    disp('A je párne číslo')
end
```

```
if rem(A,2) == 0
    disp('A je párne číslo')
else
    disp('A je nepárne číslo')
end
```

```
if A<0
    disp('číslo musí byť kladné')
elseif rem(a,2) == 0
    disp('A je párne číslo')
else
    disp('A je nepárne číslo')
end
```



Riadiace príkazy – príkaz cyklu for

- Syntax:

```
for premenná = štart : krok : koniec  
    príkazy  
end
```

Príklad: vygenerujte vektor $A(6 \times 1)$, ktorého prvok $A(1,1)=1$ a každý ďalší je dvojnásobkom predchádzajúceho

```
A(1,1) = 1  
for i = 2:6  
    A(i,1) = 2*A(i-1,1);  
end  
A
```

Zadanie: vygenerujte maticu $B(3 \times 2)$, ktorej prvky vypočítame ako pomer: index riadku/index stĺpca

Vnárание cyklov

```
for i = 1:3  
    for j = 1:2  
        B(i,j) = i/j;  
    end  
end  
B
```

Riadiace príkazy – príkaz cyklu while

používame, ak vopred nie je známy počet opakovaní tela cyklu.

Príkazy sa vykonávajú dovtedy, kým je daný výraz pravdivý. Cyklus sa nemusí vykonať ani raz, ak sa výraz vyhodnotí ako nepravdivý.

Syntax:

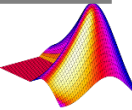
```
while výraz  
    príkazy  
end
```

Príklad: aké hodnoty bude obsahovať vektor C(1x6)?

```
while 1 % podmienka je vždy  
    pravdivá, má stále hodnotu 1  
    disp('opakujem...')  
end
```

Nekonečný cyklus
prerušíme **Ctrl+C**

```
cislo = 5  
i = 1  
while cislo ~= 0  
    C(i) = cislo + i;  
    cislo = cislo - 1;  
    i = i+1;  
end  
disp ('vektor je:')  
C
```

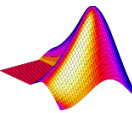


Riadiace príkazy – prepínač: switch - case - otherwise

switch sa používa pre rozlíšenie viacerých hodnôt jednej premennej alebo výrazu.

Syntax:

```
switch výraz (skalár alebo string)
  case hodnota1
    príkazy      % vykonávajú sa, ak výraz = hodnota1
  case hodnota2
    príkazy      % vykonávajú sa, ak výraz = hodnota2
  ...
  otherwise
    príkazy      % vykonávajú sa, ak výraz nie je rovný
                  ani jednej hodnote prepínača
end
```



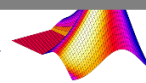
Riadiace príkazy – prepínač: switch - case - otherwise

Príklad: vygenerujte náhodné číslo z intervalu $<-1, 1>$ a k vygenerovanej hodnote vypíšte odpovedajúci text charakterizujúci číslo.

Viac hodnôt pre výraz
uvádzame do zložených
zátvoriek { }

```
y = round(rand(1)*2-1)
switch y
    case -1
        disp('hodnota je -1')
    case 0
        disp('hodnota je 0')
    case 1
        disp('hodnota je 1')
end
```

```
y = ceil(rand(1)*10)
switch y
    case 1
        disp('hodnota je 1')
    case {2,3,5}
        disp('hodnota je 2, 3 alebo 5')
    case 6
        disp('hodnota je 6')
    otherwise
        disp('ina hodnota')
end
```



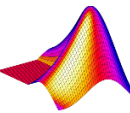
Riadiace príkazy – príkaz break

preruší vykonávanie cyklu
vytvoreného príkazmi **for** alebo **while**

Príklad:

Vygenerujte náhodné číslo
z intervalu $<0, 10>$.
Postupne od neho odpočítavajte
číslo 1, kým je výsledok väčší ako 1.
Ak je výsledok rovný štyrom
ukončíte výpočet.

```
cislo = round (rand(1)*10)
while cislo > 1
    cislo = cislo -1
    if cislo == 4
        disp ('cislo je rovne 4')
        break
    end
end
```



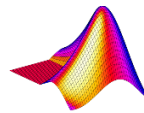
Riadiace príkazy – príkaz continue

vráti výpočet na začiatok cyklu
vytvoreného príkazmi **while** alebo **for**

Príklad:

Vygenerujte vektor (1x10)
náhodných čísel z intervalu <1, 10>.
Spočítajte v ňom počet čísel
rôzny od čísla 2.

```
A = ceil(rand(1,10)*10)
pocet = 0;
for i = 1:10
    if A(1,i) == 2
        continue
    end
    pocet = pocet + 1;
end
```



Riadiace príkazy – príkaz error

zobrazí chybové hlásenie

a preruší sa vykonávanie cyklov **for** a **while**

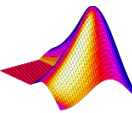
```
A = ceil(rand(1)*10)
if A >= 8
    error('Velke cislo')
end
```

Chybové hlásenie sa vypíše
v tvare:

```
??? Error using ==> prikaz_error
Velke cislo
```

```
cislo = round(rand(1)*10)
while cislo > 1
    cislo = cislo - 1
    if cislo == 4
        disp('cislo je rovne 4')
        error('Podozrive cislo')
    end
end
```

Preruší sa
vykonávanie cyklu
while



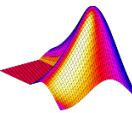
Riadiace príkazy – príkaz warning

zobrazí chybové hlásenie,
ale **program pokračuje vo vykonávaní**

```
cislo = round (rand(1)*10)
while cislo > 1
    cislo = cislo - 1
    if cislo == 4
        warning ('Kriticke cislo')
    end
end
```

Chybové hlásenie bude v tvare:

```
Warning: Kriticke cislo
> In C:\MATLAB6p5\work\subor.m at line 5
```



4. Príkazy pre užívateľský vstup – príkaz input

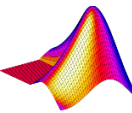
umožňuje zadať hodnotu premennej z klávesnice.

```
premenna = input('text')      % premenna je číslo  
premenna = input('text','s') % premenna je typu char
```

Príklady:

```
cislo1 = input('Zadaj prve cislo:')
```

```
meno = input('Zadaj svoje meno:', 's')
```



Príkazy pre užívateľský vstup – príkaz keyboard

slúži na prepnutie do režimu klávesnice v tele m-súboru. Príkaz **keyboard** používa sa pri ladení m-súborov (napr. umožňuje meniť parametre).

Režim klávesnice sa hlási v okne Command Window:

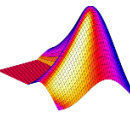
>> K

Príklad: v režime klávesnice chceme zmeniť premennú **D** (konkrétne pridať maticu **D** do súčtu):

```
a = 5;  
B = ones(3);  
keyboard  
C = a * B + D
```

```
K>> d = magic(3);  
K>> return  
>>
```

Pre ukončenie režimu klávesnice zadáme príkaz:
>> return



Príkazy pre užívateľský vstup – príkaz menu

generuje používateľské menu pre výber jednej položky.

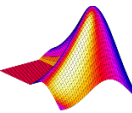
Syntax:

```
premenna = menu('nadpis','pol1','pol2',...,'poln')
```

premenna obsahuje poradové číslo vybranej položky.

Príklad:

```
k = menu('Vyber farbu:', 'červená', 'zelená', 'modrá')  
farba = ['r', 'g', 'b']  
plot(x, y, farba(k))
```



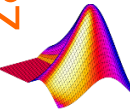
Príkazy pre užívateľský vstup – príkaz pause

preruší vykonávanie programu
a čaká na stlačenie hociktorého klávesa
alebo čaká nastavený čas

Syntax: `pause` – program zastane,
pokračuje po stlačení klávesa
`pause (n)` – n udáva počet sekúnd
prerušenia programu

`pause on` – **povoľuje** prerušenie
programu príkazom `pause`
`pause off` – **zakazuje** prerušenie
programu príkazom `pause`

dať napríklad na
začiatok programu



Linky, odporúčaná literatúra

- Hans-Petter Halvorsen **MATLAB, Part I: Introduction to MATLAB. Kap. 6.3 Functions**, University College of Southeast Norway, s. 37-40
(kniha na portáli označená ako x0)
- J. Tvrdík, V. Pavliska, P. Bujok: **Základy modelování_ v MATLABU. Kap. 6 Programování v Matlabu**, s. 44-52
(kniha na portáli označená ako x4)

