

Introduction to Matlab

Elementárna aritmetika

Elementárne operácie zoradené podľa prednosti od najnižšej po najväčšiu: Zmenu poradia prednosti

+	,	-	sčítanie, odčítanie,
*	,	/	násobenie, delenie,
^			umocňovanie.

Table 1: Elementárne operácie v Matlabe

operácii zabezpečujú zátvorky ().

Príklad

```
>> 2+3
ans =
    5
>> 3.17 - 2.77
ans =
    0.4000
>> 3/2^3
ans =
    0.3750
>> (3/2)^3
ans =
    3.3750
```

Na potlačenie výstupu sa používa ';' na konci príkazu.

Príklad

```
>> 37^21;
>>
```

Functions

Interné funkcie Matlabu začínajú malým písmenom a sú tvaru

functionname(*variable1*, *variable2*, ...)

Nasledujúce tabuľka obsahuje niektoré základné interné funkcie.

$\text{sqrt}(x)$	=	square root of x
$\text{exp}(x)$	=	e^x
$\text{log}(x)$	=	natural logarithm of x
$\text{sin}(x)$	=	sine of x in radians
$\text{cos}(x)$	=	cosine of x in radians
$\text{tan}(x)$	=	tangent of x in radians

Table 2: Niekoľko elementárnych funkcií Matlabu

x môže byť číslo, výraz, výraz obsahujúci ďalšiu funkciu, matica (definovaná nižšie) atď. Nastavením kurzora na názov funkcie a stlačením F1 je možné zistiť detaily o danej funkcii.

Príklad

```
>> sqrt(2)

ans =

    1.4142

>> sin(3.1415/2)

ans =

    1.0000

>> sqrt(sin(3.1415/4))

ans =

    0.8409
```

Premenné

Názov premennej musí začínať písmenom, nasledovať môžu potom písmená, číslice a podčiarkovníky, maximálne 63 znakov. Operátor '=' slúži na priradenie hodnoty premennej. Na výpis hodnoty premennej napíšete názov premennej. Definované premenné sa dajú použiť vo výpočtoch spolu s funkciami a aritmetickými operátormi.

Príklad

```
>> root2 = sqrt(2)

root2 =

    1.4142

>> root2

root2 =

    1.4142

>> expression = root2 + 3 * root2
```

```

expression =

    5.6569

>> sin(expression)

ans =

   -0.5862

```

Matice

Matlab je optimalizovaný na prácu s maticami. Aj samostatné číslo je v skutočnosti chápané ako 1 krát 1 matica. Na definovanie matice sa používajú zátvorky '[']'. Matice sa definujú po riadkoch; '[' začína maticu, prvky v riadkoch sú oddelené ',' alebo medzerou ' ', riadky sú oddelené bodkočiarkou ';' alebo novým riadkom (Enter). ']' ukončuje definíciu matice.

Príklad

```

>> A = [1 2 3; 4 5 6]

ans =

     1     2     3
     4     5     6

>> B = [9,8,7
        6,5,4]

ans =

     9     8     7
     6     5     4

```

Pre matice existuje viacero základných operácií:

+	,	-	klasický súčet a rozdiel matíc
*			maticové násobenie,
.*			násobenie po prvkoch,
'			transpozícia s komplexným združením
./			transpozícia bez komplexného združenia
^			mocnina matice
.^			mocnina po prvkoch
./			ľavé delenie po prvkoch; A./B obsahuje prvky A(i,j)/B(i,j)
\			nájde riešenie rovnice AX=B (Gausova metóda alebo metóda najmenších štvorcov)

Table 3: Operácie na maticiach

Cvičenie

1. Definujte si niekoľko matíc a vyskúšajte vyššie uvedené operácie.
2. Použite maticu a ko vstup do funkcií z tabuľky 2 a vysvetlite výsledok.

V matlabe existujú metódy, ktoré uľahčujú definovanie matíc špeciálneho typu. funkcia 'zeros(n)' vytvorí n krát n nulovú maticu, 'zeros(m,n)' vytvorí nulovú maticu s m riadkami and n stĺpcami. Funkcia 'ones(n)' definuje štvorcovú maticu rozmeru n obsahujúcu len 1 ako prvky a táto

funkcia má tiež verziu ktorá vygeneruje obdĺžnikovú maticu daného rozmeru. 'rand(m,n)' vytvorí (pseudo)náhodnú m krát n maticu.

Operátor ':' možno použiť v tvare 'a:b' alebo 'a:krok:b' a vygeneruje jedno riadkovú maticu čísel postupnosti od a to b s rozdielom 1 or 'krok'.

Príklad

```
>> 2:8

ans =

     2     3     4     5     6     7     8

>> 1.2:0.5:3

ans =

    1.2    1.7    2.2    2.7
```

Cvičenie

1. Definujte stĺpcový vektor (čiže maticu obsahujúci jediný stĺpec), ktorý obsahuje druhé mocniny čísel od 1 po 10.

Prístup k prvkom matice

Príkaz 'A(n)', vráti n -tý prvok matice A; v tomto prípade bez ohľadu na rozmer matice je táto chápaná ako jeden stĺpec (transformácia zhora nadol a zľava do prava).

A(i,j) vráti prvok i -teho riadku a j -teho stĺpca.

Na indexovanie je možné použiť aj iné matice, napr. A(1:3:end,:) vráti maticu pozostávajúcu z prvého, štvrtého, siedmeho, ... riadku matice A.

Cvičenie

1. Definujte Define 10 krát 10 náhodnú maticu a vytvorte z nej maticu B ktorá bude obsahovať len prvky A s párnymi súradnicami. (to generate random matrix use *rand()* or *randi()* functions.)

Logika

Dve matice rovnakého rozmeru môžeme porovnať klasickými operáciami >, <, >=, <=, ~=, ==. Porovnanie vracia maticu s prvkami 0 a 1, v závislosti od pravdivosti alebo nepravdivosti porovnania na zodpovedajúcich si súradniciach.

Príklad

```
>> A=[1 2 3;2 3 4];
>> B=[2 2 2;2 2 2];
>> A > B

ans =

     0     0     1
     0     1     1

>> A ~= B

ans =
```

$$\begin{array}{ccc} 1 & 0 & 1 \\ 0 & 1 & 1 \end{array}$$

Porovnanie vracia logickú maticu, ktorú je možné použiť na indexovanie.

Príklad

```
>> A=[1 2 3;2 3 4];
>> B=[2 2 2;2 2 2];
```

```
>> In = A ~ = B
```

In =

$$\begin{array}{ccc} 1 & 0 & 1 \\ 0 & 1 & 1 \end{array}$$

```
>> A(In) = -1
```

ans =

$$\begin{array}{ccc} -1 & 2 & -1 \\ 2 & -1 & -1 \end{array}$$

Maticu A obsahujúcu len 0 a 1 ako prvky nie je možné použiť na indexovanie, pretože nie je typu logická matica. Jej typ sa dá zmeniť na logický použitím príkazu 'logical(A)'

```
>> A=[1 2 3;2 3 4];
>> B=[1 0 1;0 1 0];
```

```
>> A(B)=-1
```

??? subscript indices must either be **real** positive integers or logicals

```
>> A(logical(B))=-1
```

ans =

$$\begin{array}{ccc} -1 & 2 & -1 \\ 2 & -1 & 4 \end{array}$$

Problémy

1. Vyriešte systém lineárnych rovníc

$$\begin{array}{r} 2x - 3y + 5z - w = 17 \\ -4x + y - 11z + 3w = -19 \\ -17x - 5y \quad + 3w = 2 \\ 5x - 3y + 12z + 12w = 9 \end{array}$$

Ak A je matica systému, B je pravá strana a X riešenie systému, vypočítajte B-A*X.

2. Nájdite riešenie systému

$$\begin{array}{r} 2x - 3y + 5z = 17 \\ -4x + y - 11z = -19 \\ -17x - 5y \quad = 2 \\ 5x - 3y + 12z = 9 \end{array}$$

Užitočná funkcia je 'size(A)', ktorá vráti rozmer matice A, t.j. Počet riadkov a počet stĺpcov v tomto poradí. Je možné definovať matice s viac ako dvoma rozmermi, typický príklad je matica RGB obrazu.

Plots

Grafy funkcií sa kreslia pomocou príkazu 'plot'. Táto funkcia vlastne kreslí závislosť medzi dvoma množinami prvkov. Vyskúšajte príkazy ako `plot([1,2,3,4])`, `plot(1:5,5:9)`, `plot(linspace(-5,5,100), sin(linspace(-5,5,100)))`. Do jedného obrazu je možné zakresiť viac funkcií. Ak definujeme 'x = linspace(-5,5,100)', čím vznikne vektor obsahujúci 100 rovnomerne rozložených čísel medzi -5 a 5, vyskúšajte 'plot(x,sin(x),x,cos(x))'. Detailnejší opis získate v dokumentácii k príkazu v MATLABE.

Obrázky

Príkaz 'imread('image_file.bmp/jpg/...')' vráti RGB maticovú reprezentáciu obrazu. Táto má tri dimenzie (ak bol obraz RGB) (použite príkaz 'size'), prvé dve sú rozmery obrazu a tretí je 3 – za každú farbu RGB jedna. Prvky tejto matice reprezentujú intenzity daných základných farieb. Existujú dve možnosti reprezentácie intenzít, buď reálnym číslom medzi 0 a 1, alebo celými číslami 0 do 255. Štandardná dvojrozmerná matica intenzít reprezentuje šedotónový obraz.

Na zobrazenie matice A (dvoj rozmerná šedotónová alebo 3 rozmerná RGB) na obrazovku sa použije príkaz 'imshow(A)'.

Príklad

```
>> A = 0:1/255:1;  
>> B = repmat(A,100,1);  
>> imshow(B)
```

funkcia 'repmat' vytvorí maticu ktorá zopakuje maticu A 100 krát v riadkoch a 1 krát v stĺpci. Výsledkom je matica so 100 riadkami 256 stĺpcami, pretože A má 1 riadok a 256 stĺpcov.

```
>> A = 0:255;  
>> B = repmat(A,100,1);  
>> imshow(uint8(B))
```

Funkcia 'uint8' zmení typ matice B, ktorá je typu double, pretože bola definovaná užívateľom na bežnamienkové 8 bitové celé číslo. Vynechaním by funkcia 'imshow' očakávala reálne hodnoty 0 až 1.

Cvičenie

1. Načítajte farebný obrázok a rozdeľte tri farebné kanály na tri rôzne matice. Zobrazte tieto kanály v ich farbách (t.j. červený kanál na červeno, atď.).
2. Načítajte farebný obraz a vytvorte jeho šedotónovú reprezentáciu. Návod: Skombinujte R, G, B kanály do jednej intenzity. Jedna z prirodzených možností je aritmetický priemer ale tento nereprezentuje skutočnú intenzitu veľmi dobre (intenzitu červenej, modrej a zelenej farby nevníma človek rovnako). Experimentujte s rôznymi váhami farebných kanálov.

Funkcia 'rgb2gray(RGBmatrix)' vracia šedotónovú maticu z RGB matice.

Príklad

```
>> RGBimg = imread('image.bmp');  
>> imshow(RGBimg);  
>> GsImg = rgb2gray(RGBimg);  
>> imshow(GsImg)
```

Problémy

1. Načítajte obraz a nasamplujte ho faktorom $1/2$, t.j. vyberte podmaticu prvkov na párnych súradniciach. Proces opakujte niekoľko krát a výsledky zobrazte. Čo pozorujete.

Na uloženie matice reprezentujúcej obraz vo forme obrazového súboru použite `'imwrite(n, 'name.ext')`, kde n je matica obrazu a *ext* typ obrazového súboru, e.g. bmp, jpg, png.

MATLAB Scripts

Vo všetkých vyššie uvedených prípadoch sme MATLAB používali ako sofistikovanú kalkulačku. Zadali sme príkaz, stlačili enter a interpret MATLABU okamžite daný príkaz vykonal. Toto nie je vždy žiaduce a často je potrebné vykonať postupnosť viacerých príkazov. Taktiež by bolo dobré mať možnosť uložiť takúto postupnosť a neskôr sa k nej vrátiť. Toto matlab umožňuje použitím skriptov.

Skript MATLABU je súbor s príponou `.m` obsahujúci postupnosť príkazov, ktorú voláme program. Program spustíme v GUI matlabu použitím veľkej zelenej šípky smerujúcej doprava. Matlab taktiež umožňuje debugovanie skriptu použitím `'breakpointov'`.

Užívateľsky definované funkcie

Takéto funkcie zapisujeme do súboru s príponou *m*, meno má byť rovnaké, ako názov funkcie.

Súbor začína definíciou hlavičky funkcie

```
function ReturnValue = FunctionName( InputValue1 , InputValue2 )
```

alebo

```
function [ ReturnValue1 , ReturnValue2 , ... ] =  
          FunctionName( InputValue1 , InputValue2 )
```

ak vraciate viac ako jednu hodnotu (maticu). Telo funkcie potom obsahuje postupnosť príkazov. Vrátenie hodnoty funkciou dosiahneme priradením tejto hodnoty do príslušnej návratovej hodnoty uvedenej v hlavičke funkcie. Napr. nasledujúca funkcia bude definovaná v súbore `'AritMean2.m'`.

```
function Result = AritMean2(A, B)  
    Result = (A + B)/2;  
end
```

Niektoré užitočné funkcie

- $\min(A)$ – Ak A je jeden riadok alebo jeden stĺpec funkcia vráti minimálny prvok v A . V opačnom prípade vracia riadok obsahujúci minimá z každého stĺpca.
- $\max(A)$ – Ak A je jeden riadok alebo jeden stĺpec funkcia vráti maximálny prvok v A . V opačnom prípade vracia riadok obsahujúci maximá z každého stĺpca.
- $\text{sum}(A)$ – Ak A je jeden riadok alebo jeden stĺpec funkcia vráti súčet prvkov v A . V opačnom prípade vracia riadok obsahujúci súčty prvkov z každého stĺpca.

Problémy

1. Napíšte a otestujte funkciu, ktorá na vstupe dostane maticu šedotónového obrazu a vráti histogram intenzít tohto obrazu. t.j vektor, ktorý má počet pixelov intenzity i v obraze na $i + 1$ -th pozícii (pozor, matlab počíta súradnice matice od 1. T.j. pozície sa počítajú od 1 a intenzity od 0). Môžete využiť funkciu `'sum'`. Histogram môžete zobraziť vo forme stĺpcového grafu príkazom `'bar'`. Alebo si naštudujte a použite funkcie `'reshape'` a `'hist'`.
2. Napíšte a otestujte funkciu, ktorej vstup je matica RGB obrazu a vráti tri histogramy, jeden za každý kanál. Môžete využiť funkciu z predchádzajúceho príkladu.

Vetvenie

```
if expression
statements
elseif expression
statements
else
statements
end
```

Ak je výraz pri 'if' pravdivý, vykonajú sa príkazy za 'if' do prvého 'else' alebo 'elseif'. Ak je výraz pri 'if' nepravdivý, vyhodnotí sa výraz pri 'elseif' atď. Ak výrazy pri 'if' a 'elseif' sú nepravdivé vykonajú sa príkazy za 'else', ak je 'else' prítomné.

Cykly

```
for index = value
statements
end
```