

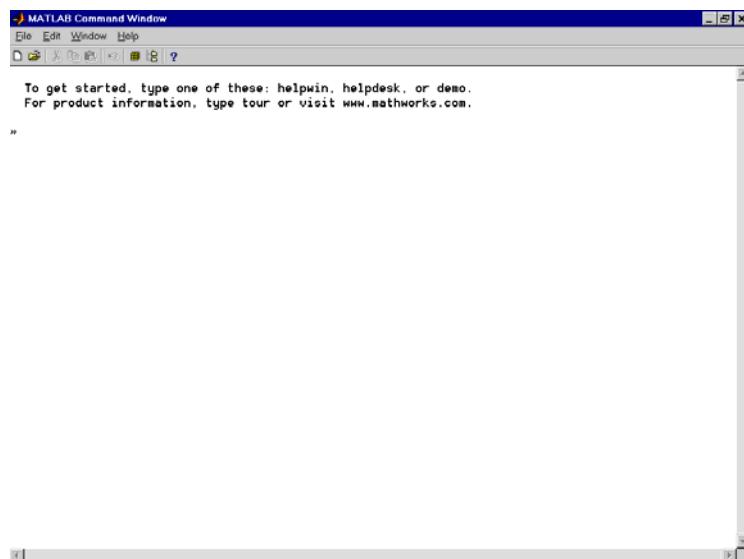
# Priručnik za MATLAB – I Deo

**MAT(rix) LAB(oratory)** predstavlja snažan softverski paket koji je i programski jezik visokog nivoa i računarski vizuelizacioni alat. Program je prvobitno bio napisan kako bi se obezbedio lak pristup softveru za rad sa matricama razvijenom u okviru projekata LINPACK i EISPACK u Argonne National Labs. Tokom godina evoluirao je u standardni alat za numerička izračunavanja na univerzitetima širom sveta i u razvojnim odeljenjima u industriji. Trenutno je aktuelna verzija 5.3.1, a postoje realizacije za više hardverskih platformi. MATLAB sadrži i niz specijalizovanih alata u obliku tzv. skript datoteka (M-datoteke) koje proširuju njegove mogućnosti u određenim klasama problema kao što su digitalna obrada signala, automatsko upravljanje, modeliranje i simulacija sistema, neuralne mreže, fazi logika, parcijalne diferencijalne jednačine itd.

## Instalacija i pokretanje MATLAB-a

MATLAB zahteva bar Windows 95 ili Windows NT na računaru sa procesorom Intel 486 i 16 MB RAM-a. Instalacija se obavlja sa CD-ROM-a. Zauzetost prostora na disku zavisi od izbora opcija prilikom instalacije i kreće se u rasponu od 30 do 270 MB. Po završenoj instalaciji u programskom meniju Windowsa imaćete stavku MATLAB koju treba aktivirati da bi se program pokrenuo. Kada se program pokrene na ekranu se pojavljuje jednostavan Windows prozor sa komandnim promptom >>.

Na vrhu prozora programa MATLAB nalazi se traka menija sa četiri stavke: File, Edit, Window i Help. Izborom stavke File pojavljuje se padajući meni sa uglavnom standardnim opcijama — New, Open, Print, Set Path, Preferences itd., koje za sada nećemo razmatrati.



Slika 1.1 Komandni prozor MATLAB-a

- Opcija **New** omogućava da napravite novu datoteku ili novi prozor za sliku.
- Opcija **Set Path** omogućava da upišete putanju kojom će MATLAB tražiti M-datoteke. U principu, ne treba ništa da uklanjate iz putanje, već samo treba da dodate odredište gde čuvate svoje M-datoteke.
- Opcija **Preferences** nudi vam mogućnost da navedete numerički format koji će MATLAB koristiti za prikazivanje izlaznog rezultata, font koji će se koristiti u

komandnom prozoru i format klipborda koji ćeete koristiti prilikom kopiranja prozora sa slikom.

**Napomena:** Pogledajte razlike između nekoliko numeričkih formata. Uočite da su `long` i `short` e formati za prikaz brojeva u pokretnom zarezu:

```
format long e; pi, 100*pi, pi/100
```

```
ans =  
3.141592653589793e+000  
ans =  
3.141592653589794e+002  
ans =  
3.141592653589794e-002
```

```
format short e; pi,100*pi,pi/100
```

```
ans =  
3.1416e+000  
ans =  
3.1416e+002  
ans =  
3.1416e-002
```

pošto su `short` i `long` fiksni decimalni formati.

```
format long; pi,100*pi,pi/100
```

```
ans =  
3.14159265358979  
ans =  
3.141592653589794e+002  
ans =  
0.03141592653590
```

```
format short; pi,100*pi,pi/10000  
ans =  
3.1416  
ans =  
314.1593  
ans =  
3.1416e-004
```

## Dobijanje pomoći

Kao i kod svakog drugog softvera, uvek je dobro pročitati priručnik za korisnika ili Help priručnik koji je obezbedio proizvođač softvera. U slučaju MATLAB-a (verzije 5.x) na raspolaganju su dva tipa Helpa, tzv. Online Help (izaberite Help Desk (HTML) u padajućem meniju) i ugrađeni Help kome se pristupa iz komandne linije ili izborom stavke Help iz padajućeg menija. Pomoć ćete obično tražiti kada vam je potrebna odgovarajuća komanda za

obavljanje konkretnog zadatka. Za početak jednostavno otkucajte `help` u komandnoj liniji i dobićete sledeće:

HELP topics:

```
matlab\general - General purpose commands.  
matlab\ops - Operators and special characters.  
matlab\lang - Programming language constructs.  
matlab\elmat - Elementary matrices and matrix manipulation.  
matlab\elfun - Elementary math functions.  
matlab\specfun - Specialized math functions.  
matlab\matfun - Matrix functions - numerical linear algebra.  
matlab\datafun - Data analysis and Fourier transforms.  
matlab\polyfun - Interpolation and polynomials.  
matlab\funfun - Function functions and ODE solvers.  
matlab\sparfun - Sparse matrices.  
matlab\graph2d - Two dimensional graphs.  
matlab\graph3d - Three dimensional graphs.  
matlab\specgraph - Specialized graphs.  
matlab\graphics - Handle Graphics.  
matlab\uitools - Graphical user interface tools.  
matlab\strfun - Character strings.  
matlab\iofun - File input/output.  
matlab\timefun - Time and dates.  
matlab\datatypes - Data types and structures.  
matlab\dde - Dynamic data exchange (DDE).  
matlab\demos - Examples and demonstrations.  
toolbox\tour - An interface to Matlab demos, installed  
Toolboxes demos, and information  
toolbox\local - Preferences.
```

For more help on directory/topic, type "help topic".

Kao primer zatražimo osnovne informacije za 2D grafiku. U komandnoj liniji upišimo sledeće:

```
help graph2d
```

Kao odgovor dobićemo sledeće:

```
Two dimensional graphs.
```

```
Elementary X-Y graphs.  
plot - Linear plot.  
loglog - Log-log scale plot.  
semilogx - Semi-log scale plot.  
semilogy - Semi-log scale plot.  
polar - Polar coordinate plot.  
plotyy - Graphs with y tick labels on the left and right.
```

Axis control.

```
axis - Control axis scaling and appearance.  
zoom - Zoom in and out on a 2-D plot.  
grid - Grid lines.  
box - Axis box.  
hold - Hold current graph.  
axes - Create axes in arbitrary positions.  
subplot - Create axes in tiled positions.
```

```
Graph annotation.  
legend - Graph legend.  
title - Graph title.  
 xlabel - X-axis label.  
 ylabel - Y-axis label.  
text - Text annotation.  
gtext - Place text with mouse.
```

```
Hardcopy and printing.  
print - Print graph or SIMULINK system; or save graph to M-  
file.  
printopt - Printer defaults.  
orient - Set paper orientation.
```

See also GRAPH3D, SPECGRAPH.

Pogledajmo sada informacije koje se dobijaju za komandu plot.

```
help plot
```

```
PLOT Linear plot.  
PLOT(X,Y) plots vector Y versus vector X. If X or Y is a  
matrix,  
then the vector is plotted versus the rows or columns of the  
matrix,  
whichever line up.
```

```
PLOT(Y) plots the columns of Y versus their index.  
If Y is complex, PLOT(Y) is equivalent to  
PLOT(real(Y),imag(Y)).  
In all other uses of PLOT, the imaginary part is ignored.
```

Various line types, plot symbols and colors may be obtained  
with

```
PLOT(X,Y,S) where S is a character string made from one  
element  
from any or all the following 3 columns:
```

```
y yellow . point - solid  
m magenta o circle : dotted  
c cyan x x-mark -. dashdot  
r red + plus -- dashed  
g green * star  
b blue s square
```

```
w white d diamond
k black v triangle (down)
^ triangle (up)
< triangle (left)
> triangle (right)
p pentagram
h hexagram
```

For example, `PLOT(X,Y,'c+:')` plots a cyan dotted line with a plus at each data point; `PLOT(X,Y,'bd')` plots blue diamond at each data point but does not draw any line.

`PLOT(X1,Y1,S1,X2,Y2,S2,X3,Y3,S3,...)` combines the plots defined by the `(X,Y,S)` triples, where the X's and Y's are vectors or matrices and the S's are strings.

For example, `PLOT(X,Y,'y-',X,Y,'go')` plots the data twice, with a solid yellow line interpolating green circles at the data points.

The `PLOT` command, if no color is specified, makes automatic use of the colors specified by the axes `ColorOrder` property. The default `ColorOrder` is listed in the table above for color systems where the default is yellow for one line, and for multiple lines, to cycle through the first six colors in the table. For monochrome systems, `PLOT` cycles over the axes `LineStyleOrder` property.

`PLOT` returns a column vector of handles to `LINE` objects, one handle per line.

The `X,Y` pairs, or `X,Y,S` triples, can be followed by parameter/value pairs to specify additional properties of the lines.

See also `SEMILOGX`, `SEMILOGY`, `LOGLOG`, `GRID`, `CLF`, `CLC`, `TITLE`, `XLABEL`, `YLABEL`, `AXIS`, `AXES`, `HOLD`, and `SUBPLOT`.

## Kako da sačuvate svoj rad

Postoje okolnosti kada morate prekinuti rad u MATLAB-u i sačuvati urađeno kako biste kasnije mogli da nastavite rad ili prepravite prethodnu radnu sesiju. Ovo je najpogodnije uraditi koristeći komandu `diary`.

Otkucajte u komandnoj liniji `help diary` da biste dobili neophodna uputstva.

```
DIARY Save text of MATLAB session.  
DIARY file_name causes a copy of all subsequent terminal input  
and most of the resulting output to be written on the named  
file. DIARY OFF suspends it. DIARY ON turns it back on.  
DIARY, by itself, toggles the diary state.
```

Use the functional form of DIARY, such as `DIARY('file')`,  
when the file name is stored in a string.

**Primer:** Sačuvajte svoju radnu sesiju kao `primer1.txt` na disketu u direktorijum `podaci`:

```
diary('a:\podaci\primer1.txt')
```

## Rad sa matricama

MATLAB će najbolje savladati ako počnete da radite sa matricama. U MATLAB-u je matrica pravougaoni niz brojeva. Specijalno značenje se ponekad pridružuje matricama dimenzija  $1 \times 1$ , koje se nazivaju skalari, i matricama sa samo jednom vrstom ili kolonom, koje se nazivaju vektori. MATLAB ima i druge načine za memorisanje numeričkih i nenumeričkih podataka, ali je za početak najbolje sve posmatrati kao matricu. Operacije u MATLAB-u projektovane su tako da izgledaju što je moguće prirodnije. Dok drugi programski jezici rade samo sa po jednim brojem istovremeno, MATLAB omogućava da radite brzo i lako sa celim matricama.

Primeri:

`c = 5.66` ili `c = [5.66]` `c` je skalar ili  $1 \times 1$  matrica

`x = [ 3.5, 33.22, 24.5 ]` `x` je vektor vrsta ili  $1 \times 3$  matrica

`x1 = [ 2  
5  
3  
-1 ]` `x1` je vektor kolona ili  $4 \times 1$  matrica

`A = [ 1 2 4  
2 -2 2  
0 3 5  
5 4 9 ]` `A` je  $4 \times 3$  matrica

## Neke osnovne operacije

### Unos matrica

Matrice možete uneti u MATLAB na više različitih načina:

- Unošenje eksplisitne liste elemenata;
- Učitavanje matrica iz spoljašnjih datoteka;

- Generisanje matrica korišćenjem ugrađenih funkcija;
- Pravljenje matrica pomoću sopstvenih funkcija u M-datotekama.

### **Unošenje eksplicitne liste elemenata**

Počnite sa unošenjem matrice kao liste elemenata. Pri unosu morate voditi računa o sledećim konvencijama:

- Elementi vrste razdvajaju se blanko znacima ili zarezima;
- Kraj vrste označava se tačkom sa zarezom;
- Listu elemenata stavite u uglaste zagrade, [ ].

Upisivanjem unesite sledeću matricu:

```
A=[1 1 3;4 0 6;2,5,-1]
```

MATLAB će matricu koju ste uneli prikazati u sledećem obliku:

```
A =
1 1 3
4 0 6
2 5 -1
```

Prikaz tačno prati redosled unosa brojeva. Kada ste jednom uneli matricu, ona se automatski pamti u radnom prostoru MATLAB-a. Kada želite da radite sa matricom A, dovoljno je samo da je navedete.

Takođe, možete pritisnuti taster Enter kada upišete jednu vrstu.

```
B=[1 0 -1
0 -1 1
-1 1 0]
```

```
B =
1 0 -1
0 -1 1
-1 1 0
```

Elementi matrice mogu biti definisani algebarskim izrazima koji se nalaze na mestima pripadajućih elemenata.

Sledeći primer

```
a = [ sin(pi/2) sqrt(2) 3+4 6/3 exp(2) ]
```

definiše matricu

```
a = [ 1.0000 1.4142 7.0000 2.0000 7.3891 ]
```

### **Indeksiranje elemenata matrica**

Element u vrsti  $i$  i koloni  $j$  matrice  $A$  označava se kao  $A(i, j)$ . Na primer,  $A(3, 2)$  je broj koji se nalazi u trećoj vrsti i drugoj koloni. U ovom primeru  $A(3, 2)$  je 5.

Matrica se može definisati na više načina korišćenjem MATLAB izraza. Sledeći primer pokazuje moguće načine u slučaju vektora vrste X dimenzija 1x3, čiji su elementi  $X(1) = 2$ ,  $X(2) = 4$  i  $X(3) = -1$ .

```
x = [ 2 4 -1 ] ili x=[2 4 -1] ili x = [ 2, 4, -1 ]
```

Matrica se može definisati povećanjem dimenzija prethodno definisane matrice. Ponovno pozivanje prethodno definisane matrice X

```
x1 = [ x 5 8 ]
```

daje sledeći rezultat:

```
x1 = [ 2 4 -1 5 8 ]
```

Izraz

```
x(5) = 8
```

daje

```
x = [ 2 4 -1 0 8 ]
```

Uočavate da je vrednost 0 postavljena na mesto elementa  $x(4)$  koji nije eksplicitno definisan.

Sada možemo izračunati sumu elemenata u trećoj koloni ranije navedene matrice A upisivanjem sledećeg izraza:

```
A(1,3) + A(2,3) + A(3,3)
```

Rezultat je:

```
ans =  
8
```

Međutim, to nije najlegantniji način za sumiranje elemenata jedne kolone.

Elemente matrice možete da referencirate sa jednim indeksom, A(k). To je uobičajen način referenciranja vektora vrste i vektora kolone. Takođe, ovo može da se primeni na punu dvodimenzionalnu matricu, jer se u tom slučaju niz posmatra kao vektor kolona formiran od kolona originalne matrice. Prema tome, za naš primer, A(6) je drugi način za referenciranje vrednosti 5 smeštene u A(3,2).

Ako pokušate da koristite vrednost elementa koga nema u matrici dobićete poruku o grešci.

```
t = A(4,5)  
Index exceeds matrix dimensions.
```

Međutim, ako pokušate da unesete vrednost u element koji ne postoji u matrici, dimenzije matrice se prilagođavaju da prihvate novu vrednost:

```
x = A;  
x(3,4) = 17
```

```
X =  
1 1 3 0  
4 0 6 0  
2 5 -1 17
```

### Operator dvotačka

Dvotačka (:) predstavlja jedan od najvažnijih operatora u MATLAB-u. Pojavljuje se u nekoliko različitih oblika. Izraz

```
1:10
```

je vektor vrsta koji sadrži celobrojne vrednosti od 1 do 10.

```
1 2 3 4 5 6 7 8 9 10
```

Da biste dobili nejedinični inkrement morate navesti vrednost inkrementa. U ovom slučaju navode se tri broja (celobrojna ili racionalna) koji su razdvojeni sa dve dvotačke. Prvi i treći broj smatraju se donjom i gornjom granicom, a srednji broj je inkrement. Na primer:

```
100:-7:50
```

Dobija se:

```
100 93 86 79 72 65 58 51
```

Ako upišete sledeći izraz:

```
0:pi/4:pi
```

Dobija se:

```
0 0.7854 1.5708 2.3562 3.1416
```

Operator dvotačka može se koristiti za dobijanje vektora iz matrice. Neka je

```
x = [ 2 6 8  
      0 1 7  
     -2 5 -6 ]
```

Sada komanda

```
y = x(:,1)
```

formira vektor kolonu

```
y = [ 2  
      0  
     -2 ]
```

a

```
yy = x(:,2)
```

formira

```
YY = [ 6  
      1  
      5 ]
```

Komanda

```
z = x(1,:)
```

formira vektor vrstu

```
z = [ 2   6   8 ]
```

Operator dvotačka zgodan je za izdvajanje manjih matrica iz većih matrica. Neka je matrica  $c$ , dimenzija  $4 \times 3$ , definisana sa

```
c = [ -1  0  0  
      1  1  0  
      1 -1  0  
      0  0  2 ]
```

Tada

```
d1 = c(:,2:3)
```

formira matricu od elemenata druge i treće kolone korišćene matrice. Nova matrica je dimenzija  $4 \times 2$  i ima sledeće elemente:

```
d1 = [  0  0  
      1  0  
     -1  0  
      0  2 ]
```

Komanda

```
d2 = c(3:4,1:2)
```

formira matricu dimenzija  $2 \times 2$ , kod koje su vrste definisane trećom i četvrtom vrstom matrice  $c$ , a kolone su definisane prvom i drugom kolonom matrice  $c$ .

```
d2 = [ 1  -1  
      0   0 ]
```

Prema tome, izrazi u indeksu omogućavaju da se referenciraju delovi matrice. Na primer,

```
A(1:k,j)
```

Referencira prvih  $k$  elemenata  $j$ -te kolone matrice  $A$ . Prema tome, izraz

```
sum(A(1:3,3))
```

izračunava sumu treće kolone. Međutim, postoji bolji način za to. Sama dvotačka, po svojoj funkciji, referencira sve elemente u vrsti ili koloni matrice, a službena reč end referencira poslednju vrstu ili kolonu. Prema tome, izraz:

```
sum(A(:, end))
```

izračunava sumu elemenata u poslednjoj koloni matrice A.

```
ans =  
8
```

### Učitavanje matrica iz spoljašnjih datoteka

Komanda load učitava binarne datoteke koje sadrže matrice generisane u prethodnim sesijama u MATLAB-u ili učitava tekstualne datoteke koje sadrže numeričke podatke. Tekstualna datoteka mora imati podatke složene tabelarno, sa po jednom vrstom u redu i sa istim brojem elemenata u svakoj vrsti. Na primer, napravite tekstualnu datoteku u editoru teksta, koja sadrži sledeća četiri reda:

```
16.0      3.0      2.0      13.0  
 5.0      10.0     11.0      8.0  
 9.0      6.0       7.0     12.0  
 4.0     15.0     14.0      1.0
```

Sačuvajte datoteku pod imenom magik.dat. Zadajte sledeću komandu:

```
load magik.dat
```

MATLAB učitava datoteku i pravi promenljivu magik koja sadrži vašu matricu.

### Generisanje matrica korišćenjem ugrađenih funkcija

MATLAB ima četiri funkcije koje generišu osnovne matrice:

zeros	Svi elementi nule
ones	Svi elementi jedinice
rand	Slučajni elementi sa uniformnom raspodelom
randn	Slučajni elementi sa normalnom raspodelom

Primeri:

```
Z = zeros(2, 4)  
Z =  
    0      0      0      0  
    0      0      0      0  
  
F = 5*ones(3, 3)  
F =  
    5      5      5  
    5      5      5  
    5      5      5
```

```

N = fix(10*rand(1,10))
N =
        4         9         4         4         8         5         2         6         8         0

R = randn(4,4)
R =
    1.0668    0.2944   -0.6918   -1.4410
    0.0593   -1.3362    0.8580    0.5711
   -0.0956    0.7143    1.2540   -0.3999
   -0.8323    1.6236   -1.5937    0.6900

```

### Komanda

a = ones(4,4)

daje

```

a=[1 0 0 0
  0 1 0 0
  0 0 1 0
  0 0 0 1]

```

Međutim, ova matrica može se dobiti i pomoću funkcije eye. Zadajte sledeću komandu:

a = eye(4)

Funkcija eye može da napravi i nekvadratne matrice. Na primer,

eye(3,2)

generiše matricu

```

1 0
0 1
0 0

```

a

eye(2,3)

generiše

```

1 0 0
0 1 0

```

### Pravljenje matrica pomoću sopstvenih funkcija u M-datotekama

Svoje matrice možete napraviti korišćenjem *M-datoteka* (to su tekstualne datoteke koje sadrže MATLAB kôd). Napravite datoteku tako da sadrži iste iskaze kao da ih upisujete u komandnoj liniji MATLAB-a. Sačuvajte datoteku pod imenom koje se završava oznakom tipa datoteke .m.

Na primer, napravite datoteku koja sadrži sledećih pet redova:

```

A = [ ...
16.0      3.0      2.0     13.0

```

```

5.0      10.0     11.0      8.0
9.0      6.0       7.0      12.0
4.0      15.0     14.0      1.0 ];

```

Sačuvajte datoteku pod imenom `magik.m`. Sada iskaz

```
magik
```

čita datoteku i pravi promenljivu, `A`, koja sadrži ovu matricu.

### Brisanje vrsta i kolona

Vrste i kolone matrice možete obrisati korišćenjem para uglastih zagrada. Prvo napravimo kopiju matrice `A`:

```
X = A;
```

Želimo sada da obrišemo drugu kolonu matrice `X`. Da bismo to uradili napišimo sledeći izraz:

```
X(:, 2) = []
```

`X` se menja u:

```
X =
16      2      13
 5      11      8
 9      7      12
 4      14      1
```

Ako obrišete samo jedan element u matrici, rezultat uopšte nije matrica. Prema tome, izrazi kao što je:

```
X(1, 2) = []
```

daju kao rezultat grešku. Međutim, korišćenjem jednog indeksa briše se jedan element ili sekvenca elemenata i preostali elementi pretvaraju se u vektor vrstu. Prema tome za

```
X(2:2:10) = []
```

dobijamo

```
X =
 16      9      2      7      13      12      1
```

### Izrazi

Kao i većina drugih programskih jezika, MATLAB obezbeđuje matematičke *izraze*, ali — za razliku od većine — ovi izrazi odnose se na kompletne matrice. Izrazi se sastoje od:

- promenljivih,
- brojeva,
- operatora i

- funkcija.

## Promenljive

MATLAB ne zahteva iskaze za navođenje tipa ili dimenzionisanje. Kada MATLAB nađe na novu promenljivu, on automatski pravi promenljivu i dodeljuje joj odgovarajuću količinu memorijskog prostora. Ako promenljiva već postoji, MATLAB menja njen sadržaj i, ako je potrebno, dodeljuje joj novi memorijski prostor. Na primer, izraz

```
br_studenata = 25
```

pravi matricu dimenzija 1x1, imenuje je `br_studenata` i smešta vrednost 25 u njen jedini element.

Ime promenljive mora početi slovom, iza koga može biti bilo koji broj slova, cifara ili crtica za podvlačenje. MATLAB koristi za identifikaciju samo prvih 13 znakova iz imena promenljive. MATLAB pravi razliku između malih i velikih slova u imenima promenljivih. A i a nisu iste promenljive. Da biste videli matricu dodeljenu nekoj promenljivoj, jednostavno navedite ime promenljive.

## Brojevi

MATLAB koristi konvencionalni decimalni način zapisivanja sa opcionom decimalnom tačkom i predznakom za brojeve. *Eksponencijalni način zapisivanja* koristi slovo `e` da označi faktore skaliranja stepenom broja 10. *Imaginarni deo* kompleksnih brojeva koristi ili `i` ili `j` kao sufiks. Evo nekoliko primera ispravno napisanih brojeva:

3	-99	0.0001
9.6397238	1.60210e-20	6.02252e23
1i	-3.14159j	3e5i

Interno, svi brojevi memoriju se u formatu long koji je definisao IEEE standard za decimalne brojeve (brojeve sa pokretnim zarezom). Decimalni brojevi imaju konačnu tačnost od oko 16 značajnih cifara i konačan raspon od oko  $10^{-308}$  do  $10^{+308}$ . (Računari tipa VAX koriste drugačiji format za decimalne brojeve, ali su tačnost i raspon skoro isti.)

## Operatori

U izrazima se koriste svima dobro poznati aritmetički operatori i pravila za redosled izvršavanja.

- + Sabiranje
- Oduzimanje
- \* Množenje
- / Desno deljenje (a/b znači a:b)
- \ Levo deljenje (a\b znači b:a)
- ^ Stepenovanje
- ' Transponovanje
- ( ) Zadavanje redosleda izvršavanja

Kada jedan red koda sadrži više od jednog od ovih operatora redosled izračunavanja se obavlja prema sledećem prioritetu:

Prioritet	Operacija
1	Zagrade
2	Stepenovanje, sleva nadesno
3	Množenje i deljenje, sleva nadesno
4	Sabiranje i oduzimanje, sleva nadesno

Ova pravila primenjuju se na skalarne veličine (tj.  $1 \times 1$  matrice) na uobičajen način. Međutim, neskalarne matrice zahtevaju dodatna pravila za primenu ovih operatora, o čemu će biti reči kasnije.

Na primer,

$3 * 4$	kada se izračunava u MATLAB-u daje:
ans=12	
$4 / 5$	kada se izračunava u MATLAB-u daje:
Ans=.8000	
$4 \backslash 5$	kada se izračunava u MATLAB-u daje:
ans=1.2500	
$x = pi/2; y = sin(x)$	kada se izračunava u MATLAB-u daje:
y = 1	
$z = 0; w = exp(4*z)/5$	kada se izračunava u MATLAB-u daje:
z = .2000	

## Funkcije

MATLAB ima veliki broj standardnih elementarnih matematičkih funkcija, uključujući `abs`, `sqrt`, `exp` i `sin`. Vađenje kvadratnog korena ili logaritma negativnog broja nije greška; automatski se dobija odgovarajući kompleksni broj. Takođe, MATLAB ima niz složenih matematičkih funkcija, uključujući Besselove i gama funkcije. Većina ovih funkcija prihvata kompleksne argumente. Da biste dobili listu elementarnih matematičkih funkcija navedite u komandnoj liniji sledeće:

```
help elfun
```

Da biste dobili listu složenih matematičkih i matričnih funkcija navedite sledeće:

```
help specfun  
help elmat
```

Neke od funkcija, kao što su `sqrt` i `sin`, već su ugrađene. One su deo jezgra MATLAB-a, tako da su veoma efikasne, ali detalji implementacije izračunavanja nisu lako dostupni. Druge funkcije, kao što su `gamma` i `sinh`, implementirane su kao M-datoteke. Njihov kôd je vidljiv i čak može po potrebi da se menja.

Nekoliko specijalnih funkcija daje vrednosti za određene konstante.

pi	3.14159265
i	Imaginarna jedinica, $(-1)^{1/2}$

<i>j</i>	Isto kao <i>i</i>
<i>eps</i>	Decimalna relativna tačnost, $2^{-52}$
<i>realmin</i>	Najmanji decimalni broj, $2^{-1022}$
<i>realmax</i>	Najveći decimalni broj, $(2 - \epsilon)2^{1023}$
<i>Inf</i>	Beskonačno
<i>NaN</i>	Nije-broj

Beskonačno se dobija deljenjem nenulte vrednosti sa nulom ili izračunavanjem zgodno definisanog matematičkog izraza koji daje *prekoračenje*, tj. premašuje *realmax*. Nije-broj se dobija u slučaju izračunavanja izraza kao što su  $0/0$  ili *Inf*-*Inf* koji nemaju definisane matematičke vrednosti.

Imena funkcija nisu rezervisana. Možete poništiti bilo koju od njih novom promenljivom, takvom kao što je

```
eps = 1.e-6
```

a zatim koristiti tu vrednost u svim narednim izračunavanjima. Originalna funkcija može se povratiti sa

```
clear eps
```

### Izrazi

Već ste imali priliku da vidite nekoliko primera MATLAB izraza. Evo još nekoliko sa odgovarajućim rezultatima.

```

rho = (1+sqrt(5))/2
rho =
    1.6180

a = abs(3+4i)
a =
    5

z = sqrt(besselk(4/3,rho-i))
z =
    0.3730+ 0.3214i

huge = exp(log(realmax))
huge =
    1.7977e+308

toobig = pi*huge
toobig =
    Inf
```

### Matrična izračunavanja

Pošto matrice sadrže nizove elemenata, a ne samo jedan element (izuzetak je skalarna 1x1 matrica), zakoni komutacije, asocijacija i distribucije iz aritmetike ne važe uvek. Ipak, niz značajnih pravila važiće u matričnoj algebri i MATLAB-u kada se radi sa neskalarnim veličinama.

### Sabiranje i oduzimanje matrica

Samo matrice **istog reda** mogu se sabirati i oduzimati. Kada se dve matrice istog reda sabiraju ili oduzimaju u matričnoj algebri, pojedinačni elementi se sabiraju ili oduzimaju. Prema tome, zakon distribucije važi.

$$A + B = B + A \quad \text{and} \quad A - B = B - A$$

Ako je  $C = A + B$ , tada je svaki element  $C(i, j) = A(i, j) + B(i, j)$ .

Definišimo  $A$  i  $B$  na sledeći način:

$$\begin{aligned} A &= [1 \ 2 \ 3; \ 3 \ 3 \ 3; \ 5 \ 3 \ 1] \\ B &= [2 \ -3 \ 4; \ 2 \ -2 \ 2; \ 0 \ 4 \ 0] \end{aligned}$$

Zatim uočimo da

$$C = A + B \quad \text{and} \quad C = B + A$$

daje

$$\begin{matrix} C = \\ \begin{matrix} 3 & -1 & 7 \\ 5 & 1 & 5 \\ 5 & 7 & 1 \end{matrix} \end{matrix}$$

Definišimo sada vektor vrstu

$$x = [3 \ 5 \ 7]$$

i vektor kolonu

$$y = [4; \ -1; \ -3]$$

Uočite da operacija

$$z = x + y$$

nije dozvoljena, pošto matrice nisu istog reda. ( $x$  je matrica 1x3, a  $y$  je matrica 3x1.) Dozvoljeno je sabiranje proizvoljnog broja 1x1 matrica ili skalara i važe pravila iz aritmetike. Dozvoljeno je sabiranje dva vektora samo ako su obe vektor vrsta (matrica 1xn) ili vektor kolona (matrica nx1)

Kvadratne matrice mogu se uvek sabirati i oduzimati ukoliko su istog reda. Kvadratna matrica dimenzija 4x4 ne može se sabirati sa kvadratnom matricom dimenzija 3x3, zato što nisu istog reda, mada su obe kvadratne matrice.

## Množenje matrica

Množenje matrica je mnogo složenije nego aritmetičko množenje, zato što svaka matrica sadrži brojne elemente. Podsetimo se da je kod množenja vektora postojanje niza elemenata u vektoru dovelo do dva koncepta množenja, skalarnog i vektorskog proizvoda. Množenje matrica takođe ima niz svojih specijalnih pravila.

Kod množenja matrica, elementi proizvoda, matrica C, dve matrice A i B, izračunavaju se po formuli

$$C(i, j) = A(i, k) * B(k, j)$$

Da bi se matrice mogle množiti broj kolona prve ili leve matrice (A) mora biti jednak broju vrsta druge ili desne matrice (B). Proizvod, matrica C, ima broj vrsta koji je jednak broju vrsta prve (leve) matrice (A), a broj kolona koji je jednak broju kolona druge (desne) matrice (B). Jasno je da ne mora biti  $A^*B$  jednako  $B^*A$ . Takođe, jasno je da  $A^*B$  i  $B^*A$  postoje samo kod kvadratnih matrica.

Zadajmo matrice

$$\begin{aligned} A &= \\ 1 &\quad 1 \quad 3 \\ 4 &\quad 0 \quad 6 \\ 2 &\quad 5 \quad -1 \end{aligned}$$

i

$$\begin{aligned} B &= \\ 1 &\quad 0 \quad -1 \\ 0 &\quad -1 \quad 1 \\ -1 &\quad 1 \quad 0 \end{aligned}$$

Matrični proizvod  $C = A^*B$  daje sledeće:

$$\begin{aligned} C &= \\ -2 &\quad 2 \quad 0 \\ -2 &\quad 6 \quad -4 \\ 3 &\quad -6 \quad 3 \end{aligned}$$

Izračunajmo sada prouzvod matrica x i y.

$$\begin{aligned} x &= [2 \quad 3 ; \quad 4 \quad -1 ; \quad 0 \quad 7] ; \\ y &= [5 \quad -6 \quad 7 \quad 2 ; \quad 1 \quad 2 \quad 3 \quad 6] ; \end{aligned}$$

Prvo, uočite da proizvod matrica x i y postoji, pošto matrica x ima isti broj kolona (2) koliko matrica y ima vrsta (2). Takođe, uočite da proizvod  $y^*x$  **ne postoji!** Ako je proizvod matrica c, tada je ona matrica dimenzija 3x4.

**Napomena:** Proizvod skalara i matrice jeste matrica kod koje je svaki element dobijen množenjem odgovarajućeg elementa matrice i skalara.

## Operacije tipa element-po-element

Sabiranje i oduzimanje matrica obuhvata sabiranje i oduzimanje pojedinačnih elemenata matrica. Ponekad je potrebno jednostavno pomnožiti ili podeliti svaki element matrice sa

odgovarajućim elementom druge matrice. Ove operacije se u MATLAB-u zovu *array operations*, a ovde će se koristiti termin operacije element-po-element. Ove operacije se izvršavaju kada se ispred operatora nalazi znak tačka (.). Prema tome

- $a . * b$  množi svaki element u  $a$  sa odgovarajućim elementom iz  $b$
- $a . / b$  deli svaki element u  $a$  sa odgovarajućim elementom iz  $b$
- $a . \ b$  deli svaki element u  $b$  sa odgovarajućim elementom iz  $a$
- $a . ^ b$  stepenuje svaki element u  $a$  sa odgovarajućim elementom iz  $b$

Neka su matrice  $G$  i  $H$  definisane sledećim elementima:

```
G = [ 1   3   5; 2   4   6];
H = [-4   0   3; 1   9   8];
```

Tada je

```
G .* H = [ -4    0   15
            2   36   48 ]
```

### Transponovanje matrice

Transponovana matrica dobija se razmenom vrsta i kolona polazne matrice. Za ovu operaciju MATLAB koristi operator apostrof (').

Neka je matrica  $A$

```
A =
1 1 3
4 0 6
2 5 -1
```

tada transponovanje matrice,  $A'$ , daje sledeće:

```
ans =
1 4 2
1 0 5
3 6 -1
```

Determinanta matrice  $\det(A)$  je

```
ans =
46
```

a rang matrice  $\text{rank}(A)$  je

```
ans = 3
```

Kao što vidite ovo su jednostavne komande u MATLAB-u.

Postoji više ugrađenih komandi za standardna matrična izračunavanja. Na primer,

karakteristične vrednosti matrice  $D$  izračunavaju se korišćenjem komande `eig(D)`

```
ans =  
3.0000  
3.0000  
0.0000
```

Ako želite da izračunate karakteristične vektore koristićete komandu `[P, LAMDA] = eig(D)`

```
P =  
0.80341805013121 0.14555446880815 0.57735026918963  
-0.27565515744340 -0.76855767567667 0.57735026918963  
-0.52776289268782 0.62300320686851 0.57735026918963
```

```
LAMDA =  
3 0 0  
0 3 0  
0 0 0
```

gde je  $P$  matrica karakterističnih vektora, a  $LAMDA$  dijagonalna matrica karakterističnih vrednosti, tako da važi  $A = P(LAMDA)P^{-1}$ .

## Algebarske operacije

Ponekad je potrebno da se obave algebarske operacije nad svim elementima matrice. Kao što je već rečeno, ovo se u MATLAB-u postiže korišćenjem operatora `+, -, *, ^` kojima prethodi tačka. Na primer, ako je `AA=[-1 2 0; 4 -2 5; 3 2 1]`

```
AA =  
-1 2 0  
4 -2 5  
3 2 1
```

tada `AA.^2` daje kao rezultat kvadrat svakog elementa matrice `AA`.

```
ans =  
1 4 0  
16 4 25  
9 4 1
```

Neka je `BB=[9 8 7; 6 5 4; 3 2 1]`

```
BB =  
9 8 7  
6 5 4  
3 2 1
```

i izračunajmo proizvod `AA.*BB`

```
ans =  
-9 16 0  
24 -10 20  
9 4 1
```

kod koga se odgovarajuće stavke množe.

Kvadriranje ranije definisane matrice  $B$ ,  $D=B^2$ , daje sledeće:

```
D =
2 -1 -1
-1 2 -1
-1 -1 2
```

Uočite efekat eksponencijalne operacije sa decimalnom tačkom  $DD=B.^2$

```
DD =
1 0 1
0 1 1
1 1 0
```

Kao sledeći primer izračunajmo niz vrednosti funkcije  $(\sin x)/x$ . Sledeće komande prave niz ekvidistantnih tačaka od 0.1 do 1 za argument  $x$ , vrednosti sinusa ( $y$ ) za svaku vrednost  $x$  u nizu i količnik ovih vrednosti.

```
x=linspace(.1,1,9),y=sin(x), z=y./x

x =
Columns 1 through 7
0.1000 0.2125 0.3250 0.4375 0.5500 0.6625 0.7750
Columns 8 through 9
0.8875 1.0000
y =
Columns 1 through 7
0.0998 0.2109 0.3193 0.4237 0.5227 0.6151 0.6997
Columns 8 through 9
0.7755 0.8415
z =
Columns 1 through 7
0.9983 0.9925 0.9825 0.9684 0.9503 0.9284 0.9029
Columns 8 through 9
0.8738 0.8415
```

### Skalarni proizvod dva vektora

Skalarni ili unutrašnji proizvod dva vektora vrste,  $G1$  i  $G2$ , izračunava se na niže navedeni način. Napravite vektore vrste dekomponovanjem matrice  $G$ . Neka su

```
G = [ 1   3   5; 2   4   6];
G1 = G(1,:);
G2 = G(2,:)
```

Tada je skalarni proizvod  $1 \times 3$  vektora vrste  $G1$  i  $1 \times 3$  vektora vrste  $G2$

```
G1 * G2' = 44
```

Proverite ovaj rezultat u MATLAB-u.

Ako su oba vektora vektori kolone, tada se može izračunati skalarni proizvod matričnim množenjem transponovanog prvog vektora kolone sa drugim vektorkom kolonom, tj. operacijom kod koje se matrica  $1 \times n$  množi sa matricom  $n \times 1$ .

Da zaključimo. Skalarni proizvod mora uvek biti proizvod vektora vrste i vektora kolone.

### Spoljašnji proizvod vektora

Spoljašnji proizvod dva vektora vrste, na primer ranije definisanih  $G_1$  i  $G_2$ , jeste  $G_1' * G_2$ . (Uočite da je  $G_1'$   $3 \times 1$ , a  $G_2$   $1 \times 3$ .) Rezulat je kvadratna matrica.

Ako su oba vektora vektori kolone, spoljašnji proizvod dobija se množenjem jednog vektora kolone sa transponovanim drugim vektorom kolonom.

**Napomena:** Nemojte mešati spoljašnji proizvod sa vektorskim proizvodom u mehanici.

## Zadaci za vežbanje

I. U sledećim zadacima odredite veličinu navedenih matrica i izračunajte matrice, a potom to što ste dobili proverite izvršavanjem potrebnih opracija u MATLAB-u. Da biste mogli rešiti zadatke morate ih rešavati navedenim redosledom.

1.  $a = [1, 0, 0, 0, 0, 1]$
2.  $b = [2; 4; 6; 10]$
3.  $c = [5 \quad 3 \quad 5; \quad 6 \quad 2 \quad -3]$
4.  $d = \begin{bmatrix} 3 & 4 \\ 5 & 7 \\ 9 & 10 \end{bmatrix}$
5.  $e = [3 \quad 5 \quad 10 \quad 0; \quad 0 \quad 0 \quad \dots]$   
 $\quad \quad \quad 0 \quad 3; \quad 3 \quad 9 \quad 9 \quad 8]$
6.  $t = [4 \quad 24 \quad 9]$   
 $q = [t \quad 0 \quad t]$
7.  $x = [3 \quad 6]$   
 $y = [d; x]$   
 $z = [x; d]$
8.  $r = [c; \quad x, 5]$
9.  $v = [c(2,1); \quad b]$
10.  $a(2,1) = -3$

II. Neka je definisana matrica  $g$ , dimenzija  $5 \times 4$

$$g = \begin{bmatrix} 0.6 & 1.5 & 2.3 & -0.5 \\ 8.2 & 0.5 & -0.1 & -2.0 \\ 5.7 & 8.2 & 9.0 & 1.5 \\ 0.5 & 0.5 & 2.4 & 0.5 \\ 1.2 & -2.3 & -4.5 & 0.5 \end{bmatrix}$$

Odredite sadržaj i dimenzije sledećih matrica i proverite svoje rezultate u MATLAB-u.

```
1. a = g(:,2)
2. b = g(4,:)
3. c = [10:15]
4. d = [4:9;1:6]
5. e = [-5,5]
6. f= [1.0:-.2:0.0]
7. t1 = g(4:5,1:3)
8. t2 = g(1:2:5,:)
```

## Priručnik za MATLAB (II deo)

### Rešavanje sistema linearih jednačina

Jednu od najuobičajenijih primena matrične algebre predstavlja rešavanje sistema linearih jednačina. Neka imamo skup od  $n$  jednačina sa  $n$  nepoznatih  $x_1, x_2, \dots, x_n$ .

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \dots + a_{2n}x_n = b_2$$

$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \dots + a_{3n}x_n = b_3$$

.

.

.

.

$$a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \dots + a_{nn}x_n = b_n$$

Matrični oblik ovog sistema linearih jednačina je

$$[a][x] = [b]$$

gde je

$$\begin{array}{ccccccccc} a_{11} & a_{12} & a_{13} & \dots & a_{1n} & & x_1 & & b_1 \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} & & x_2 & & b_2 \\ [a] = a_{31} & a_{32} & a_{33} & \dots & a_{3n} & [x] = & x_3 & [b] = & b_3 \\ \cdot & \cdot & \cdot & \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & & \cdot & & \cdot \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} & & x_n & & b_n \end{array}$$

Klasično rešenje ove matrične jednačine može se dobiti na osnovu definicije inverzne matrice. Inverzna matrica je ona matrica koja kada se pomnoži sa originalnom matricom kao rezultat daje jediničnu matricu. Prema tome, važi sledeće

$$a^{-1} a = I$$

gde  $a^{-1}$  označava inverznu matricu, a  $I$  označava jediničnu matricu koja je istog reda kao matrica  $a$ . Ako je  $a$  matrica poznatih koeficijenata, a  $b$  vektor kolona slobodnih članova, problem se sastoji u nalaženju  $n$  nepoznatih vrednosti koje zadovoljavaju sistem linearih jednačina. Množenjem prethodne jednačine ( $[a][x] = [b]$ ) inverznom matricom  $a^{-1}$ , ali sa leve strane, dobija se

$$[a^{-1}][a][x] = [a^{-1}][b]$$

ili

$$[x] = [a^{-1}][b]$$

Prema tome, ako nađemo inverznu matricu matrice koeficijenata, tada proizvod te matrice i matrice slobodnih članova daje nepoznatu matricu  $\mathbf{x}$ .

MATLAB za izračunavanje inverzne matrice koristi komandu `inv()`.

Kao primer rešavanja sistema linearnih jednačina korišćenjem inverzne matrice razmotrimo sledeći sistem jednačina.

$$x_1 - 4x_2 + 3x_3 = -7$$

$$3x_1 + x_2 - 2x_3 = 14$$

$$2x_1 + x_2 + x_3 = 5$$

U MATLAB-u definišite dve matrice `[a]` i `[b]`.

```
a = [ 1 -4 3; 3 1 -2; 2 1 1];
b = [ -7; 14; 5];
```

Vektor rešenja  $\mathbf{x}$  je

```
x = inv(a)*b
```

odnosno

```
x = [ 3
      1
      -2 ]
```

**Napomena:** Pri rešavanju praktičnih inženjerskih problema ne koristi se rešavanje pomoću inverzne matrice jer je računski veoma intenzivno.

### Levo i desno matrično deljenje u MATLAB-u

Metod koji se preporučuje za rešavanje velikih sistema linearnih jednačina jeste metod zamene unazad, pošto se jednačine premeštaju tako da se prvo dobija nepoznata  $x_n$ , zatim  $x_{n-1}$  i tako redom sve do  $x_1$ . Prema tome, kod metode zamene unazad jednačine se ne rešavaju istovremeno, već sekvenčno. Na primer, ranije navedeni sistem jednačina može se preuređiti u sledeći

$$x_1 - 4x_2 + 3x_3 = -7$$

$$13x_2 - 11x_3 = 35$$

$$(34/13)x_3 = (68/13)$$

Kada je sistem u ovom obliku, treća jednačina daje rešenje  $x_3 = 2$ . Zamenom vrednosti za  $x_3$  u drugoj jednačini dobija se rešenje za  $x_2 = 1$ , a zamenom  $x_3$  i  $x_2$  u prvoj jednačini dobija se rešenje  $x_1 = 1$ . Metod zamene unazad koristi se u MATLAB-u kada zadajemo komande „levo deljenje” i „desno deljenje”.

Da vas podsetimo, u teoriji matrica deljenje matrica nije definisano. Sintaksa „levo deljenje” ili „desno deljenje” jednostavno uvodi metode zamene promenljivih unazad da bi se dobio vektor rešenja sistema linearnih jednačina. Da li će se primeniti „levo deljenje” (`\`) ili „desno deljenje” (`/`) zavisi od toga kako je jednačina postavljena.

## Levo deljenje

Ako je matrična jednačina data u obliku

$$[a] [x] = [b]$$

tada  $n \times n$  matrica  $[a]$  množi sa leve strane  $n \times 1$  matricu  $[x]$  i dobija se  $n \times 1$  matrica  $[b]$ . Rešenje za  $x$  dobija se korišćenjem operacije levo deljenje.

$$[x] = [a] \setminus [b]$$

Ako je matrična jednačina data u tom obliku, korišćenje desnog deljenja za dobijanje vektora rešenja  $x$  daće kao rezultat poruku o greški.

Ranije su date matrice  $[a]$  i  $[b]$  u formatu matrične jednačine koji je pogodan za levo deljenje.

$$\begin{aligned} a &= [1 \ -4 \ 3; 3 \ 1 \ -2; 2 \ 1 \ 1]; \\ b &= [-7; 14; 5]; \end{aligned}$$

Sada zadajte komandu

$$x1 = a \setminus b$$

i dobija se rezulat

$$x1 = \begin{bmatrix} 3 \\ 1 \\ -2 \end{bmatrix}$$

što je isto kao ono ranije dobijeno pomoću inverzne matrice.

## Desno deljenje

Matrični oblik u kojem se desno deljenje upotrebljava manje je uobičajen pri rešavanju sistema linearnih jednačina, ali ne i manje uspešan. Desno deljenje se uvodi kada su jednačine napisane u obliku

$$[x] [A] = [B]$$

Uočite da su kod ovog oblika  $[x]$  i  $[B]$  vektori vrste, a ne vektori kolone kao ranije. Ovaj oblik predstavlja  $1 \times n$  matricu  $(x)$  koja sa leve strane množi  $n \times n$  matricu  $(A)$ , a kao rezultat se dobija  $1 \times n$  matrica  $(B)$ . Vratimo se opet sistemu jednačina

$$x_1 - 4x_2 + 3x_3 = -7$$

$$3x_1 + x_2 - 2x_3 = 14$$

$$2x_1 + x_2 + x_3 = 5$$

Ove jednačine mogu se napisati u sledećem matričnom obliku

$$[x] [A] = [B],$$

ako su

$$x = [x_1 \ x_2 \ x_3] \quad i \quad B = [-7 \ 14 \ 5]$$

a

$$[A] = \begin{bmatrix} 1 & 3 & 2 \\ -4 & 1 & 1 \\ 3 & -2 & 1 \end{bmatrix}$$

Uočite da je matrica  $[A]$  transponovana matrica  $[a]$ , tj.

$$A = a'$$

Pošto imamo ovako definisane  $A$  i  $B$ , rešenje za  $x$  može se dobiti desnim deljenjem.

$$x = B/A$$

Rešenje je

$$x = \begin{bmatrix} 3 \\ 1 \\ -2 \end{bmatrix}$$

## Statističke funkcije

MATLAB raspolaže nizom komandi za obavljanje jednostavnih statističkih izračunavanja. Sada ćemo dati objašnjenja za neke od tih funkcija.

<code>max(x)</code>	Daje maksimalnu vrednost elementa vektora ili, ako je $x$ matrica, daje vektor vrstu čiji su elementi maksimalne vrednosti iz odgovarajućih kolona matrice.
<code>min(x)</code>	Daje minimum od $x$ (analogno <code>max(x)</code> ).
<code>mean(x)</code>	Daje srednjokvadratnu vrednost elemenata vektora ili, ako je $x$ matrica, daje vektor vrstu čiji su elementi srednjokvadratne vrednosti elemenata iz svake kolone matrice.
<code>median(x)</code>	Isto kao <code>mean(x)</code> , samo daje medijana vrednost.
<code>sum(x)</code>	Daje sumu elemenata vektora ili, ako je $x$ matrica, daje sumu elemenata iz svake respektivne kolone matrice.
<code>prod(x)</code>	Isto kao <code>sum(x)</code> , samo daje proizvod elemenata.
<code>std(x)</code>	Daje standardnu devijaciju elemenata vektora ili, ako je $x$ matrica, daje vektor vrstu čiji su elementi standardna devijacija svake kolone matrice.
<code>sort(x)</code>	Sortira vrednosti u vektoru $x$ ili kolonama matrice i ređa ih u rastućem redosledu. Uočite da će ova komanda poremetiti odnose koji možda postoje između elemenata u vrsti matrice $x$ .
<code>hist(x)</code>	Crta histogram elemenata vektora $x$ . Deset podeljaka se označava na osnovu maksimalne i minimalne vrednosti.
<code>hist(x, n)</code>	Crta histogram sa $n$ podeljaka skaliranih između maksimalne i minimalne vrednosti elemenata.
<code>hist((x(:, 2)))</code>	Crta histogram elemenata druge kolone matrice $x$ .

Započnimo vežbanje formiranjem matrice dimenzija 12x3, koja predstavlja vremenske serije dva merenja temperature.

Otvorite neki editor i unesite niže navedene podatke. Sačuvajte ASCII datoteku pod imenom vremtemp.dat.

Vreme (sec)	Temp-T1 (K)	Temp-T2 (K)
0.0	306	125
1.0	305	121
2.0	312	123
3.0	309	122
4.0	308	124
5.0	299	129
6.0	311	122
7.0	303	122
8.0	306	123
9.0	303	127
10.0	306	124
11.0	304	123

Učitajte datoteku u MATLAB. Podatke ćete sada referencirati kao matricu vremtemp.

Zadajte komande iz gornje tabele i posmatrajte dobijene rezultate. Uočite da je, kada je argument vremtemp, rezultat vektor dimenzija 1x3. Na primer, komanda

```
M = max(vremtemp)
```

daje

```
M = 11 312 129
```

Ako je argument kolona matrice, komanda identificuje konkretnu željenu kolonu. Na primer, komanda

```
M2 = max(vremtemp(:, 2))
```

daje

```
M2 = 312
```

Ako se traži varijansa skupa podataka iz kolone matrice, standardna devijacija se kvadrira. Komanda

```
T1_var = (std(vremtemp(:, 2)))^2
```

daje

```
T1_var = 13.2727
```

Ako se traži standardna devijacija podataka matrice

```
STDDEV = std(vremtemp)
```

dobiјa se

```
STDDEV = 3.6056 3.6432 2.3012
```

Uočite da komanda

```
VAR = STDDEV^2
```

nije prihvatljiva; međutim, komanda

```
VAR = STDDEV.^2
```

je prihvatljiva i daje sledeći rezultat:

```
VAR = 13.0000 13.2727 5.2955
```

## Aproksimiranje numeričkih podataka analitičkom funkcijom

U praksi često imamo potrebu da za skup vrednosti parova  $(x, y)$  pronađemo jednačinu ili funkciju koja ih aproksimira. Procedura koju ćemo ovde razmatrati može se generalno klasifikovati u dve metode, metodu interpoliranja i metodu najmanjih kvadrata.

### Metoda interpoliranja podataka

Pretpostavimo da imate parove podataka  $(x, y)$  koji su dati tabelarno. Prirodno je očekivati da postoji neka funkcionalna zavisnost između ovih podataka. Svakom paru vrednosti  $(x_i, y_i)$  odgovaraće po jedna tačka na dijagramu. Kriva linija koja prolazi kroz sve tačke na ovom dijagramu zove se interpolaciona kriva, a sve tačke kroz koje kriva prolazi zovu se i čvorne tačke interpolacione krive.

Interpolaciona kriva koja prolazi kroz  $n+1$  tačaka na grafu dijagrama može se uvek aproksimirati jednačinom  $y = F(x)$  u obliku parabole  $n$ -tog reda, jer se zamenom vrednosti iz tabele dobija sistem od  $n+1$  jednačina za određivanje  $n+1$  koeficijenata  $a_0, a_1, \dots, a_n$  parabole  $n$ -tog reda. Opšti oblik polinoma može se napisati na sledeći način

$$y = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots + a_nx^n$$

a odgovarajući sistem jednačina je

$$\begin{aligned} y_1 &= a_0 + a_1x_1 + a_2x_1^2 + a_3x_1^3 + \dots + a_nx_1^n \\ y_2 &= a_0 + a_1x_2 + a_2x_2^2 + a_3x_2^3 + \dots + a_nx_2^n \\ y_3 &= a_0 + a_1x_3 + a_2x_3^2 + a_3x_3^3 + \dots + a_nx_3^n \\ &\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\ &\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\ &\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \end{aligned}$$

$$y_{n+1} = a_0 + a_1x_{n+1} + a_2x_{n+1}^2 + a_3x_{n+1}^3 + \dots + a_nx_{n+1}^n$$

Ovo je rešiv sistem od  $n+1$  jednačina sa  $n+1$  nepoznatih. Nepoznati su koeficijenti  $a_0, a_1, \dots, a_n$ .

Na osnovu matrične algebre i desnog (ili levog) deljenja u MATLAB-u se mogu vrlo jednostavno dobiti nepoznati koeficijenti.

Sistem jednačina može se pogodno izraziti u obliku matrične jednačine

```
[y] = [X] [a]
```

gde su

$$\begin{array}{ccccccccc} y_1 & & & & & & & & a_1 \\ & 1 & & x_1 & & x_1^2 & \dots & x_1^{n+1} & \end{array}$$

$$\begin{array}{ccccccccc}
 & y_2 & & 1 & x_2 & x_{22} \dots x_{2n+1} & & a_2 \\
 [y] = & & [x] = & & & & [a] = & \\
 & y_3 & & 1 & x_3 & x_{32} \dots x_{3n+1} & & a_3 \\
 & \cdot & & \cdot & \cdot & \cdot & & \cdot \\
 & \cdot & & \cdot & \cdot & \cdot & & \cdot \\
 & \cdot & & \cdot & \cdot & \cdot & & \cdot \\
 & y_{n+1} & & 1 & x_{n+1} & x_{n+12} \dots x_{n+1n+1} & & a_{n+1}
 \end{array}$$

Uočite da desna strana matrične jednačine mora biti proizvod od „n+1”x „n+1” kvadratne matrice i „n+1”x1 matrice kolone! U MATLAB-u se korišćenje Gaussove metode eliminacije naziva levo deljenje ili

$$[a] = [x] \setminus [y]$$

Korišćenjem ove metode mogu se dobiti koeficijenti polinoma n-tog stepena koji tačno prolaze kroz „n+1” tačku podataka.

Ako imate veliki skup podataka, korišćenje polinoma n-tog stepena nije dobar izbor. Polinomi stepena većeg od pet ili šest često se problematično ponašaju između tačaka podataka, mada tačno prolaze kroz svaku tačku. Za primer razmotrimo niže navedene podatke.

x	y
2	4
3	3
4	5
5	4
6	7
7	5
8	7
9	10
10	9

Imamo skup podataka za devet tačaka. Posmatranjem podataka uočava se da sa porastom x raste i y. Međutim, ispostavlja se da se to dešava na nelinearan način. Utvrđimo šta podaci daju. U MATLAB-u formirajmo dva vektora ovih podataka.

$$x^9 = [2:1:10];$$

$$y^9 = [4 3 5 4 7 5 7 10 9];$$

Pošto imamo devet parova podataka, može se formirati polinom osmog stepena.

$$y = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots + a_8x^8$$

Da bi se pronašli nepoznati koeficijenti potrebno je definisati vektor kolonu y

$$y = y^9'$$

kao i matricu x

$$x = [\text{ones}(1, 9); x^9; x^9.^2; x^9.^3; x^9.^4; x^9.^5; x^9.^6; x^9.^7; x^9.^8]'$$

Uočite da je x definisano korišćenjem transponovanja, funkcije `ones()` i element-po-element operatora „.^”. Tako definisani x i y zadovoljavaju jednačinu

```
[X] [a] = [y]
```

Da biste u MATLAB-u izračunali koeficijente matrice a zadajte sledeću komandu

```
a = X\y
```

Dobija se sledeće rešenje

```
a = [ 1.0e+003*
      3.8140
     -6.6204
      4.7831
     -1.8859
      .4457
     - .0649
      .0057
     - .0003
      .0000 ]
```

Uočite da je deveti koeficijent a(8) nula. U stvari, on je konačne veličine, a pojavljuje se kao nula zato što MATLAB prikazuje samo četiri značajne decimale. Da bi se prikazali koeficijenti sa više značajnih cifara potrebno je zadati sledeću komandu

```
format long
a
```

i format je promenjen na 15 značajnih cifara. Jasno je da a(8) nije nula, ono je mali broj zato što množi broj x koji se diže na osmi stepen.

### Metoda najmanjih kvadrata

Možda bi kod prehodnog primera polinom trećeg stepena bio zadovoljavajući izbor; međutim, polinom trećeg stepena ima četiri nepoznata koeficijenta, a imamo na raspolaganju devet tačaka. Koje četiri tačke od devet da izaberemo? Metoda najmanjih kvadrata taj problem rešava tako što pronalazi „srednju” ili najbolju krivu koja sadrži informacije od svake tačke podataka. Razmatranje algoritma premašuje granice ovog priručnika i čitaoci se upućuju na usko stručnu literaturu iz oblasti numeričke analize. Zadržaćemo se na polinomijalnom obliku i odrediti rezultat za polinom trećeg stepena.

Ako želite da podatke aproksimirate kvadratnom ili linearном jednačinom, lako ćete izabrati redukovane matrice koje daju tražene koeficijente. Prema tome, pronađimo koeficijente za jednačinu

```
y = a0 + a1x + a2x^2 + a3x^3
```

Četiri nepoznata koeficijenta se izračunavaju iz

```
a = X \ B
```

gde su

```
a0          m   'xi 'xi2 'xi3           'yi
```

$$\begin{array}{lll}
 [a] = \begin{matrix} a_1 & 'xi & 'xi_2 & 'xi_3 & 'xi_4 \\ & a_2 & 'xi_2 & 'xi_3 & 'xi_4 & 'xi_5 \\ & a_3 & & 'xi_3 & 'xi_4 & 'xi_5 & 'xi_6 \end{matrix} & [X] = \begin{matrix} 'xi_{iy} \\ 'xi_{2yi} \\ 'xi_{3yi} \end{matrix} & [B] = \begin{matrix} 'xi_{iy} \\ 'xi_{2yi} \\ 'xi_{3yi} \end{matrix}
 \end{array}$$

Sumira se na intervalu od 1 do  $m$ , gde je  $m$  broj parova podataka  $(x_i, y_i)$ . Uočite da kod metode najmanjih kvadrata morate imati bar jedan podatak više nego što imate nepoznatih koeficijenata, tj. mora biti  $m > n+1$ .

MATLAB ima funkciju za izračunavanje polinoma (jednačine) koja aproksimira podatke metodom najmanjih kvadrata. Ta komanda je

```
polyfit(x, y, n)
```

gde su  $x$  i  $y$  vektori podataka, a  $n$  red polinoma kojim se vrši aproksimacija metodom najmanjih kvadrata. Komanda `polyfit` kao rezultat vraća vektor čiji su elementi koeficijenti polinoma. Elementi vektora su u obrnutom redosledu u odnosu na redosled koji imaju kao koeficijenti kod polinoma, tj. prvi je koeficijent najvišeg reda, a poslednji je koeficijent najnižeg reda.

## Zadaci za vežbanje

Rešite sledeće sisteme linearnih jednačina korišćenjem inverzne matrice i levog i desnog deljenja.

a)  $r + s + t + w = 4$

$$2r - s + w = 2$$

$$3r + s - t - w = 2$$

$$r - 2s - 3t + w = -3$$

b)  $2x_1 + x_2 - 4x_3 + 6x_4 + 3x_5 - 2x_6 = 16$

$$-x_1 + 2x_2 + 3x_3 + 5x_4 - 2x_5 = -7$$

$$x_1 - 2x_2 - 5x_3 + 3x_4 + 2x_5 + x_6 = 1$$

$$4x_1 + 3x_2 - 2x_3 + 2x_4 + x_6 = -1$$

$$3x_1 + x_2 - x_3 + 4x_4 + 3x_5 + 6x_6 = -11$$

$$5x_1 + 2x_2 - 2x_3 + 3x_4 + x_5 + x_6 = 5$$

## 2D grafika

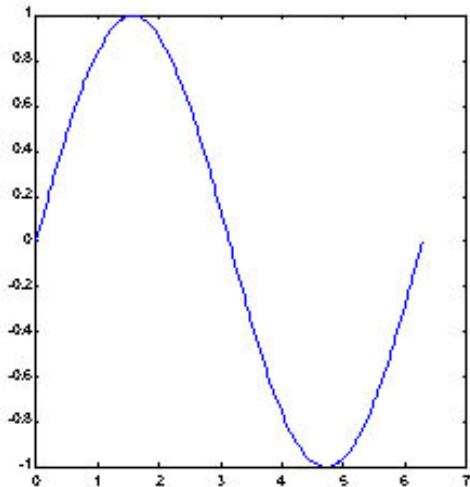
MATLAB raspolaže bogatim repertoarom komandi za prikazivanje vektora i matrica u grafičkom obliku, kao i za označavanje i štampanje tih dijagrama.

### Crtanje dijagrama

Komanda `plot` ima različite oblike u zavisnosti od ulaznih argumenata. Ako je `y` vektor, tada `plot(y)` crta stepenasti linearni dijagram elemenata vektora `y` u zavisnosti od rednog broja elementa vektora `y`. Ako navedete dva vektora kao argumente, `plot(x, y)` crta dijagram zavisnosti vektora `y` od vektora `x`.

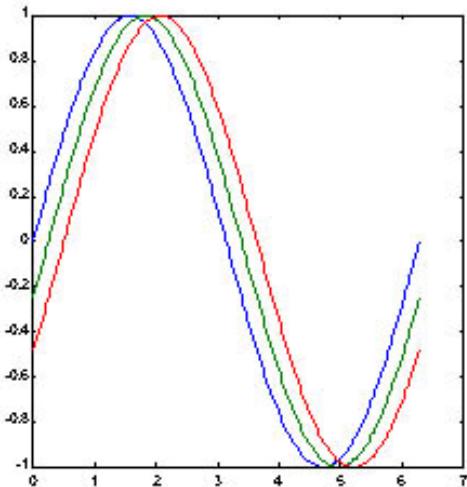
Na primer, da bi MATLAB nacrtao dijagram vrednosti sinusne funkcije za ugao u intervalu od 0 do  $2\pi$  treba da zadate sledeće komande:

```
t = 0:pi/100:2*pi;
y = sin(t);
plot(t,y)
```



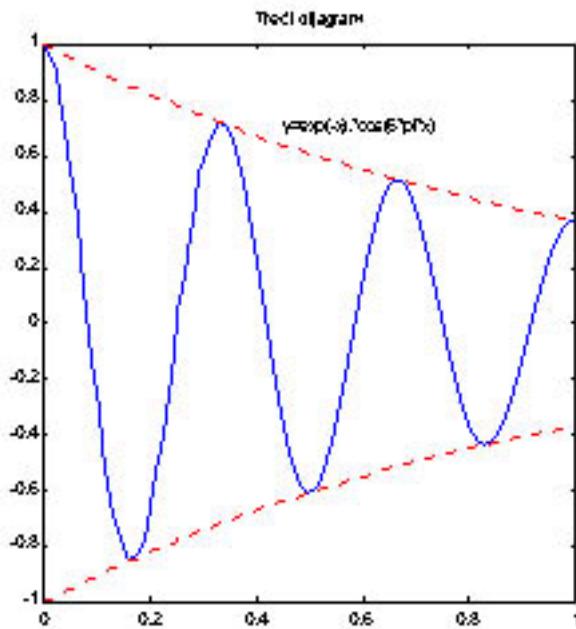
Ako imate više parova `x-y`, jednim pozivom komande `plot` biće iscrtani svi dijagrami. MATLAB automatski kruži po elementima liste boja (koju korisnik može prilagoditi svojim potrebama) da bi svakom skupu podataka dodelio drugu boju za prikazivanje. Na primer, sledeće komande crtaju tri funkcije koje zavise od argumenta `t`, a svaka kriva ima drugačiju boju:

```
y2 = sin(t-.25);
y3 = sin(t-.5);
plot(t,y,t,y2,t,y3)
```



Sada ćemo nacrtati eksponencijalno opadajuću funkciju na intervalu  $(0,1)$ , a zatim i njenu ovojnicu. Dijagramu ćemo dati ime „Treći dijagram”, koristeći komandu `title`. Takođe, pomoću komande `text` dodaćemo nisku znakova na određeno mesto na dijagramu. Evo niza komandi:

```
x=linspace(0,1,100);
y=exp(-x).*cos(6*pi*x);
plot(x,y)
x=linspace(0,1,100);
y=exp(-x).*cos(6*pi*x);
w=exp(-x);z=-w;
plot(x,y,'b-',x,w,'r:',x,z,'r:'),title('Treći dijagram')
text(.45,.7,'y=exp(-x).*cos(6*pi*x)')
```



Boju, vrstu linije i markere kojima će biti iscrtana kriva možete zadati sledećom komandom:

```
plot(x,y,'color_style_marker')
```

*color\_style\_marker* jeste niska od 1, 2 ili 3 znaka koji su međusobno razdvojeni apostrofom, a odnose se, redom, na boju, vrstu linije i vrstu markera.

- Oznake za boju su 'c', 'm', 'y', 'r', 'g', 'b', 'w' i 'k'. One odgovaraju početnim slovima engleskih naziva za boje, tj. cyan, magenta, yellow, red, green, blue, white i black.
- Oznake za vrste linija su '-' za punu liniju, '--' za isprekidanu liniju, ':' za tačkastu liniju, '-.' za crta-tačka liniju i 'none', koja označava da nema linije.
- Markeri koji se obično koriste su '+', 'o', '\*' i 'x'.

Na primer, komanda

```
plot(x, y, 'y:+')
```

crti žutu tačkastu liniju i postavlja marker u obliku znaka + kod svake tačke skupa podataka. Ako zadate marker i none za vrstu linije, MATLAB će crtati samo markere.

### Prozori figure

Komanda `plot` automatski otvara novi prozor za sliku, ako on već ne postoji na ekranu. Ako prozor za sliku postoji na ekranu, komanda `plot` podrazumeva da ga treba koristiti. Da biste otvorili novi prozor za sliku i da bi on postao tekući prozor za sliku zadajte komandu `figure`. Da biste zadali da neki od postojećih prozora za sliku postane tekući zadajte sledeću komandu

```
figure(n)
```

gde je `n` redni broj prozora za sliku. Rezultati dejstva potonjih grafičkih komandi prikazuju se u tom prozoru.

### Dodavanje dijagrama na postojeći dijagram

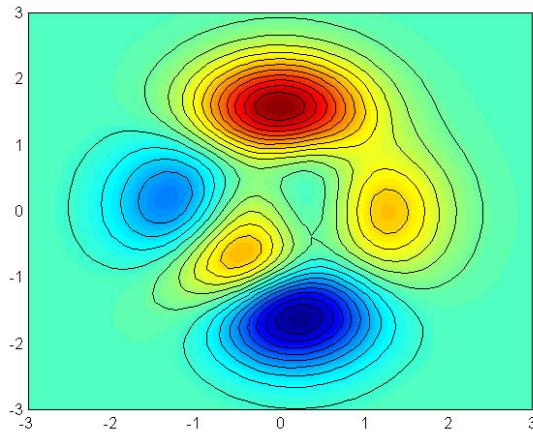
Komanda `hold` omogućava da na postojeći dijagram dodate novi dijagram. Kada zadate komandu

```
hold on
```

MATLAB ne uklanja postojeći dijagram; on dodaje nove podatke postojećem dijagramu, normalizujući ga ako je potrebno. Na primer, sledeći niz komandi prvo crta konturni dijagram funkcije `peaks`, a zatim superponira kvazikolorni dijagram iste funkcije:

```
[x, y, z] = peaks;
contour(x, y, z, 20, 'k')
hold on
pcolor(x, y, z)
shading interp
```

Komanda `hold on` omogućava da se dijagram `pcolor` kombinuje sa dijagramom `contour` na istoj slici.



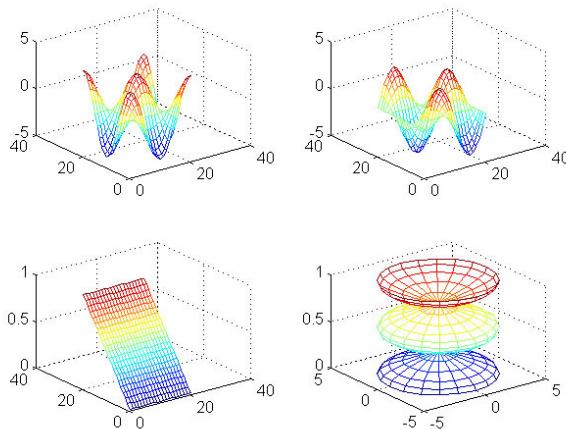
### Komanda subplot

Komanda `subplot` omogućava da prikažete više slika dijagrama u istom prozoru ili da ih štampate na istom listu papira. Komanda

```
subplot(m, n, p)
```

deli prozor slike u matricu, tj. na  $m \times n$  malih oblasti, a zatim bira  $p$ -tu oblast za tekući prozor slike. Crteži su numerisani po redovima, sleva nadesno, odozgo nadole. Na primer, sledeći niz komandi crta dijagrame u četiri različite oblasti jednog prozora za sliku:

```
t = 0:pi/10:2*pi;
[X,Y,Z] = cylinder(4*cos(t));
subplot(2,2,1)
mesh(X)
subplot(2,2,2); mesh(Y)
subplot(2,2,3); mesh(Z)
subplot(2,2,4); mesh(X,Y,Z)
```



### Imaginarni i kompleksni podaci

Kada su argumenti komande `plot` kompleksni brojevi, imaginarni deo se zanemaruje, izuzev kada je komandi `plot` dat jedan kompleksni argument. Za ovaj specijalni slučaj komanda crta dijagram zavisnosti realnog dela od imaginarnog dela. Prema tome,

```
plot(Z)
```

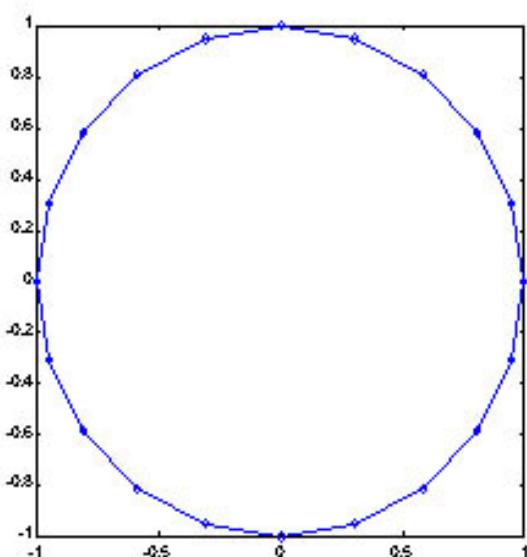
gde je Z kompleksni vektor ili matrica, ekvivalentno je sa

```
plot(real(Z), imag(Z))
```

Na primer, sledeće komande

```
t = 0:pi/10:2*pi;  
plot(exp(i*t), '-o')
```

crtaju 20-strani poligon sa malim kružićima na temenima.



### Definisanje koordinatnih osa

Komanda `axis` ima više opcija za prilagođavanje normalizacije, orijentacije i proporcije dijagrama.

Prirodno, MATLAB pronalazi minimalnu i maksimalnu vrednost u podacima i prema tome određuje dimenzije dijagrama, kao i označavanje po koordinatnim osama. Komanda `axis` ima prioritet u odnosu na tako ustanovljene vrednosti i postavlja svoje argumente za granice koordinatnih osa.

```
axis([xmin xmax ymin ymax])
```

Takođe, `axis` prihvata brojne ključne reči za podešavanje osa. Na primer, komanda

```
axis square
```

definiše da su x i y osa iste dužine, a komanda

```
axis equal
```

definiše da su podeoci na obe ose iste veličine. Prema tome, komanda

```
plot(exp(i*t))
```

zajedno sa komandom axis square ili axis equal pretvara ovalni oblik u odgovarajući krug.

Komanda

```
axis auto
```

vraća normalizaciju osa na podrazumevani, automatski režim.

Komanda

```
axis on
```

isključuje označavanje osa i podelu na osama.

Komanda

```
axis off
```

isključuje označavanje osa i podelu po osama.

Komanda

```
grid off
```

isključuje linije pomoćne mreže, a komanda

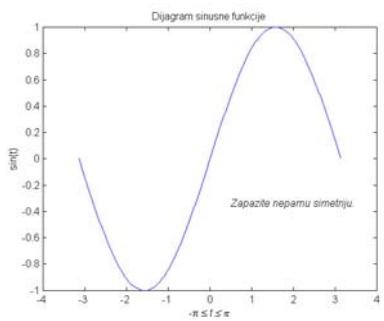
```
grid on
```

ponovo ih uključuje.

### Označavanje osa i dijagrama

Komande xlabel, ylabel i zlabel dodaju oznake x-, y- i z-osi. Komanda title dodaje naziv dijagrama na vrh prozora za sliku, a komanda text ubacuje tekst bilo gde na sliku. Može se koristiti deo TeX-ove sintakse da bi se na dijagramu pojavila grčka slova, matematički simboli i alternativni fontovi. U sledećem primeru koristi se oznaka \leq za  $\leq$ , \pi za  $\pi$  i \it za ispisivanje kurzivom.

```
t = -pi:pi/100:pi;
y = sin(t);
plot(t,y)
axis([-pi pi -1 1])
xlabel('-\pi \leq t \leq \pi')
ylabel('sin(t)')
title('Dijagram sinusne funkcije')
text(1,-1/3,'Zapazite neparnu simetriju.')'
```



# Priručnik za MATLAB (IV deo)

Pripremio Dragan Marković

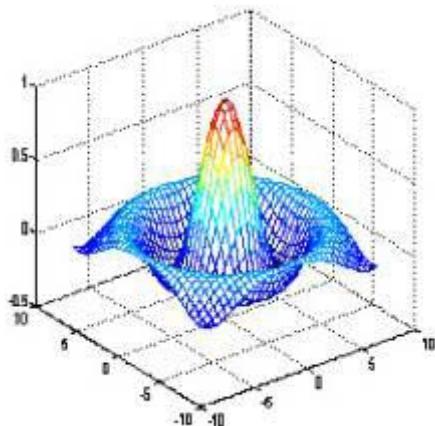
## Vizuelizacija funkcija dve promenljive

MATLAB definiše površinu sa z-koordinatama iznad mreže u x-y ravni korišćenjem pravih linija za povezivanje susednih tačaka. Funkcije mesh i surf prikazuju površine u tri dimenzije. Funkcija mesh daje tzv. žičani prikaz površina tako što boji samo linije koje povezuju definisane tačke. Funkcija surf prikazuje u boji i spojne linije i male elementarne površine (pločice; engl. *facets*).

Da biste prikazali neke funkcije dve promenljive možete koristiti više tehnika. Možete napraviti mrežu tačaka iz oblasti definisanosti funkcije, tj. matrice X i Y koje sadrže vrste i kolone koji se ponavljaju, respektivno. Zatim koristite te matrice da biste izračunali vrednosti funkcije i grafički je prikazali. Funkcija meshgrid transformiše domen (oblast definisanosti) zadat vektorima x i y u matrice X i Y koje se koriste za izračunavanje funkcija dve promenljive. Vrste X predstavljaju kopije vektora x, a kolone Y kopije vektora y.

Sada ćemo izračunati i prikazati funkciju **sinc**, tj. **sin(r)/r**, između pravaca x i y:

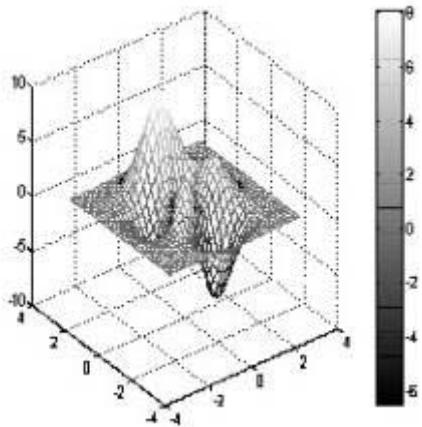
```
[X,Y] = meshgrid(-8:.5:8);
R = sqrt(X.^2 + Y.^2) + eps;
Z = sin(R)./R;
mesh(X,Y,Z)
```



U ovom primeru R je odstojanje od koordinatnog početka, koji je i centar matrice. Vrednost eps dodaje se da bi se izbegla neodređenost tipa 0/0 u koordinatnom početku.

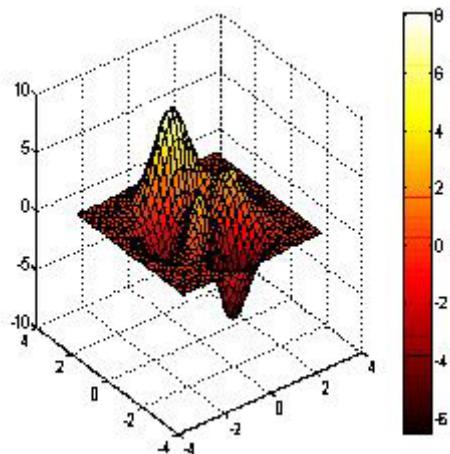
Na sledećoj slici koristi se siva skala kao kolorna mapa.

```
x=-3:.2:3;
y=-3:.2:3;
[X,Y]=meshgrid(x,y);
Z=peaks(X,Y);
colormap gray
mesh(X,Y,Z)
colorbar
```



Ako želite da obojite male elementarne površine tako da izgledaju kao zamrljana okna staklenog prozora koristite komandu surf i kolornu mapu hot.

```
x=-3:.2:3;
y=-3:.2:3;
[X,Y]=meshgrid(x,y);
Z=peaks(X,Y);
colormap hot
surf(X,Y,Z)
colorbar
```



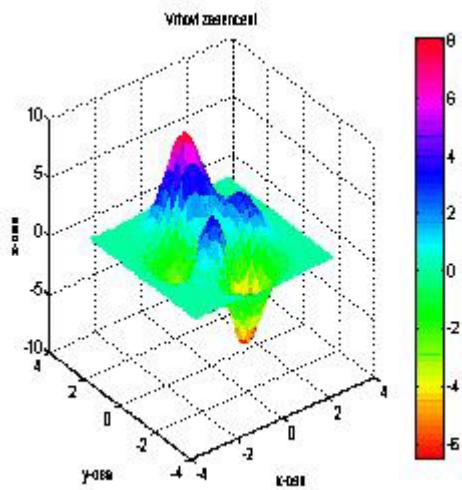
MATLAB, takođe, raspolaže jednoličnim i interpoliranim senčenjem. Kod ovih senčenja uklanjuju se crne linije. Kod jednoličnog senčenja elementarne površine zadržavaju svoju individualnu boju (koja je definisana z-koordinatom), dok se kod interpoliranog senčenja svakoj elementarnoj površini dodeljuje interpolirana nijansa (što inače zahteva prilično izračunavanje). Slede primeri oba senčenja.

```
x=-3:.2:3;
y=-3:.2:3;
[X,Y]=meshgrid(x,y);
Z=peaks(X,Y);
colormap hsv
surf(X,Y,Z)
shading flat
```

```

xlabel('x-osa'), ylabel('y-osa'), zlabel('z-osa')
title('Vrhovi zasenceni')
colorbar

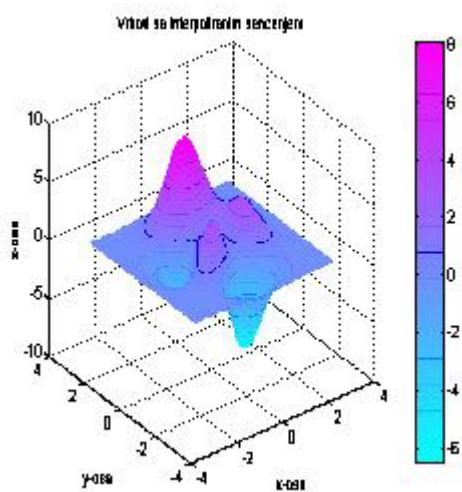
```



```

x=-3:.2:3;
y=-3:.2:3;
[X,Y]=meshgrid(x,y);
Z=peaks(X,Y);
colormap cool
surf(X,Y,Z)
shading interp
xlabel('x-osa'), ylabel('y-osa'), zlabel('z-osa')
title('Vrhovi sa interpoliranim sencenjem')
colorbar

```



Ovim primerima samo su delimično ilustrovane mogućnosti MATLAB-a u oblasti vizuelizacije. MATLAB omogućava da napravite svoju kolornu mapu, zadate svjetlosne efekte, postavite tačku posmatranja (gledišta) itd. O tome će više biti reči u nekom od narednih tekstova.

# **Priručnik za MATLAB (V deo)**

**Pripremio Dragan Marković**

---

U ovom tekstu upoznaćemo se sa komandama CD, LOAD, SAVE i PRINT.

## **Komanda CD**

Komandom CD možete da promenite tekući radni direktorijum tako što ćete navesti putanju do direktorijuma gde se nalaze datoteke sa podacima. Na primer:

CD DRAGAN

Ako navedete samo komandu CD, bez ikakvog argumenta, MATLAB ispisuje naziv tekućeg direktorijuma. Ako navedete sledeću komandu:

CD ..

MATLAB prelazi u direktorijum koji se nalazi iznad tekućeg direktorijuma na hijerarhijskom stablu volumena.

## **Komanda LOAD**

Komandom LOAD učitavate sadržaj datoteke u tekući radni prostor MATLAB-a. Datoteka po tipu može biti binarna (.mat; izvorni MATLAB-ov format) ili ASCII. Međutim, MATLAB prihvata i podatke koji se nalaze u Excelovim radnim tabelama.

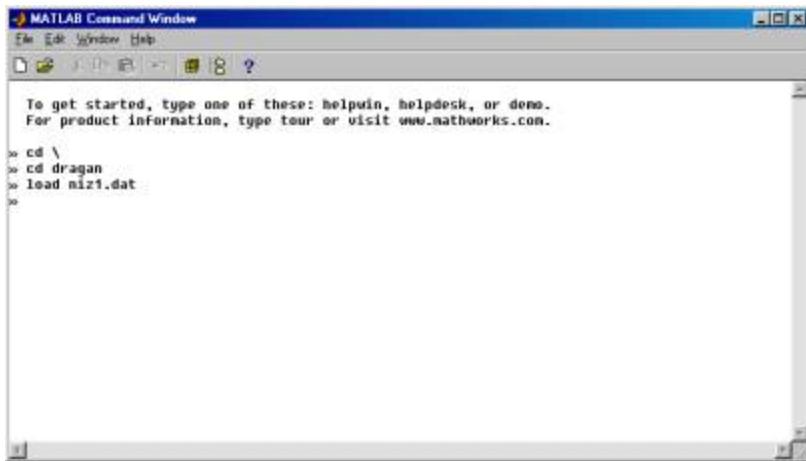
Ako na raspolaganju imate neki skup podataka, dobijen merenjima ili anketiranjem ispitanika, koji se nalazi u nekoj datoteci, možete ga učitati u MATLAB radi dalje obrade.

Prepostavimo, prvo, da imate podatke koji se nalaze u datoteci u tzv. čistom ASCII formatu. Takvu datoteku možete napraviti upisivanjem podataka u bilo kom programu za uređivanje ASCII teksta (na primer, MS DOS Editor). Podaci su u tzv. ravnoj (engl. flat) datoteci, tj. poređani su u kolone, a kolone su razdvojene blanko znakom ili sa više blanko znakova.

**Napomena:** ASCII datoteka je datoteka koja sadrži znakove (podatke) u ASCII formatu, tj. formatu koji je nezavisан od MATLAB-a ili bilo kog drugog programa.

Recimo da u direktorijumu C:\DRAGAN imamo ASCII datoteku NIZ1.DAT u kojoj se nalaze dve kolone sa podacima. Da bismo pristupili tim podacima u MATLAB-u treba da navedemo sledeće komande:

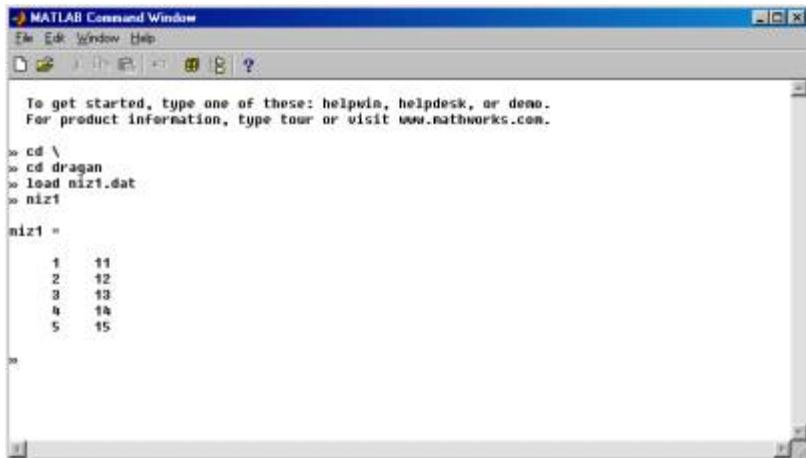
```
cd \
?d dragan
load niz1.dat
```



```
MATLAB Command Window
File Edit Window Help
File Edit Window Help
>> To get started, type one of these: helpwin, helpdesk, or demo.
For product information, type tour or visit www.mathworks.com.

>> cd \
>> cd dragan
>> load niz1.dat
>>
```

Podaci su sada na raspolaganju u MATLAB-ovom radnom okruženju, što možemo proveriti ako u komandnoj liniji navedemo naziv datoteke, koji će MATLAB prihvati kao naziv promenljive i ispisati podatke na ekranu.



```
MATLAB Command Window
File Edit Window Help
File Edit Window Help
>> To get started, type one of these: helpwin, helpdesk, or demo.
For product information, type tour or visit www.mathworks.com.

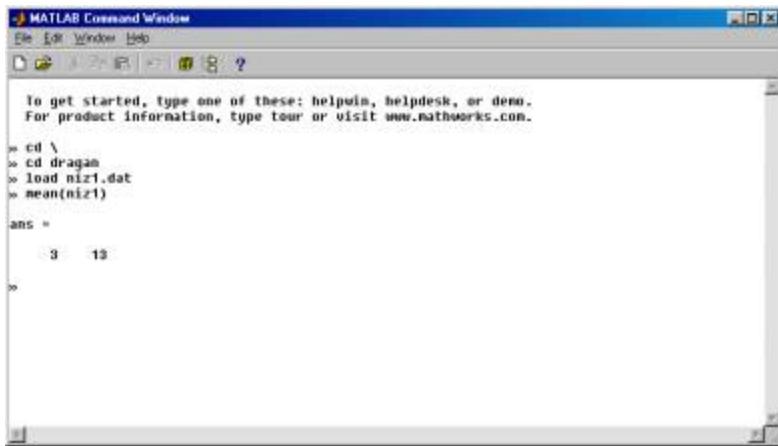
>> cd \
>> cd dragan
>> load niz1.dat
>> niz1

niz1 =
    1    11
    2    12
    3    13
    4    14
    5    15

>>
```

Sada možemo, na primer, izračunati srednju vrednost podataka u kolonama. U komandnoj liniji navedimo sledeću komandu:

```
mean(niz1)
```



The screenshot shows the MATLAB Command Window. The window title is "MATLAB Command Window". The menu bar includes "File", "Edit", "Window", and "Help". The toolbar has icons for file operations like Open, Save, and Help. The command history and output area contains the following text:

```
To get started, type one of these: helpwin, helpdesk, or demo.  
For product information, type tour or visit www.mathworks.com.  
  
=> cd '\  
>> cd dragam  
>> load niz1.dat  
>> mean(niz1)  
  
ans =  
    3    13  
  
>>
```

Evo objašnjenja kako se učitavaju podaci iz Excelove radne tabele ili iz tabele u Wordu. Otvorite u Excelu radnu tabelu sa podacima. Podrazumeva se da ste MATLAB ostavili otvoren. Obeležite mišem grupu celija sa kojom želite da radite, tj. koju ćete kopirati u MATLAB. Sada u meniju Edit izaberite stavku Copy. Vratite se u MATLAB. U komandnoj liniji upišite sledeće:

```
P=[];
```

Posle znaka ; nemojte da pritisnete taster Enter! Postavite pokazivač miša između zagrada i pritisnite levi taster na mišu, a zatim u meniju Edit izaberite stavku Paste. Potom postavite pokazivač miša iza znaka ; i pritisnite taster Enter.

**Napomena:** Kada radite sa velikim skupovima podataka često ćete na kraju reda koji se odnosi na podatke stavljati znak ;. Ovim znakom saopštavate MATALB-u da ne prikazuje te podatke na ekranu.

Sada možete da nastavite obradu podataka učitanih na ovaj način.

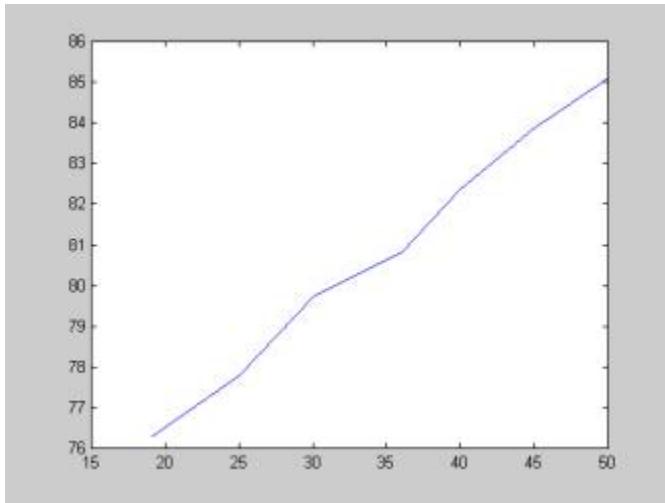
## Komanda SAVE

Komandom SAVE možete da sačuvate rezultate radne sesije u binarnu datoteku (.mat). Kasnije te podatke možete ponovo komandom LOAD učitati u MATLAB kako biste nastavili obradu ili vizuelno prikazivanje.

Prepostavimo da ste merenjem električnog otpora nekog provodnika pri raznim temperaturama dobili rezultate koje ste upisali u datoteku podaci.dat. Kada ove rezultate prikažete grafički videćete da je među njima skoro linearan zavisnost tipa

$$R = a_0 + a_1 * t$$

gde su  $a_0$  i  $a_1$  konstante koje mogu da se odrede na osnovu merenja i MATLAB-ove funkcije polyfit.



Funkcija polyfit izračunava koeficijente polinoma koji "najbolje" povezuje podatke po metodi najmanjih kvadrata. Funkcija zahteva tri argumenta:  $x$ ,  $y$  i  $n$ ;  $x$  i  $y$  su vektori podataka, a  $n$  je red polinoma. Prema tome za najbolju linearu ( $n=1$ ) zavisnost treba da napišemo sledeće:

```
cd \
cd dragan
load podaci.dat
polyfit(podaci(:,1),podaci(:,2),1)
```

**Napomena:**  $\text{podaci}(:,1)$  adresira prvu kolonu podataka iz datoteke, a  $\text{podaci}(:,2)$  drugu.

MATLAB daje koeficijente u opadajućem redosledu stepena polinoma. Prema tome, u ovom slučaju "najbolja" linearna aproksimacija je:

$$R = 70,76 + 0,288*t$$

Ako bismo želeli da koeficijenti budu elementi vektora koef tada bi trebalo da napišemo sledeće:

```
koef=polyfit(podaci(:,1),podaci(:,2),1)
```

Sada možemo iskoristiti komandu SAVE da sačuvamo koeficijente u datoteci koef.mat tako što ćemo napisati:

```
save koef
```

Datoteka se nalazi u tekućem radnom direktorijumu i kasnije je možemo učitati radi daljeg korišćenja.

### **Komanda PRINT**

Stavka Print iz menija File i komanda print ravnopravno se koriste za štampanje slika stvorenih u MATLAB-u. Komanda Print iz menija ima standardni okvir za dijalog u kome birate standardne opcije za štampanje. Komanda print nudi više slobode u izboru tipa izlaza i omogućava da upravljate štampanjem iz m-datoteke. Rezultat može biti direktno poslat na štampač ili memorisan u navedenu datoteku. Na raspolaganju su razni formati.

Na primer, sledećom komandom memorišemo sadržaj tekućeg prozora slike u datoteku otpornost.eps, kao sliku u boji u formatu PostScript Level 2:

```
print -depsc2 otpornost.eps
```

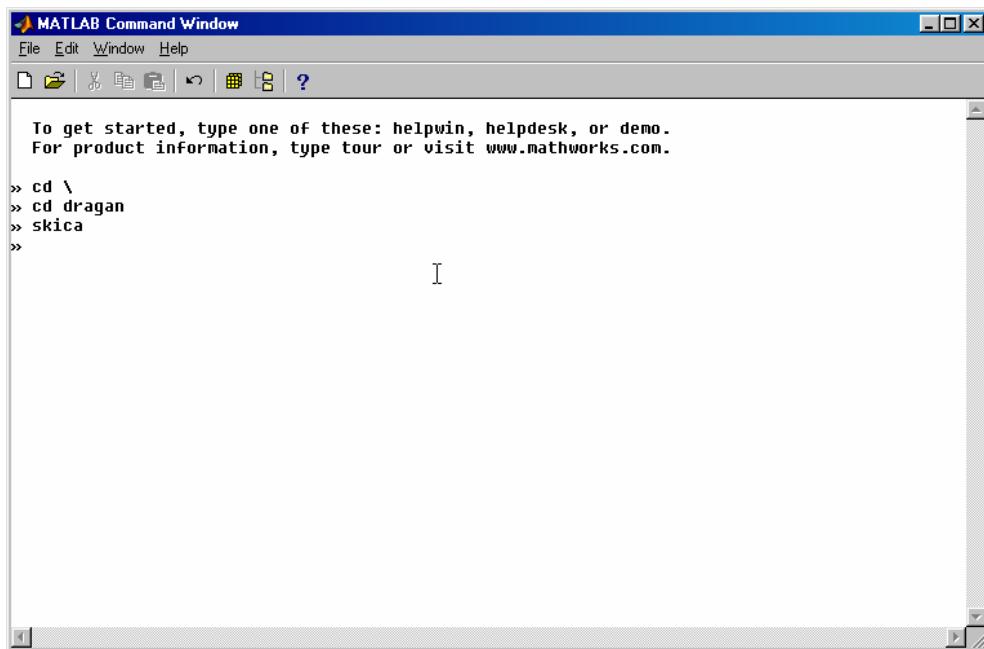
## MATLAB (6)

### M-datoteke

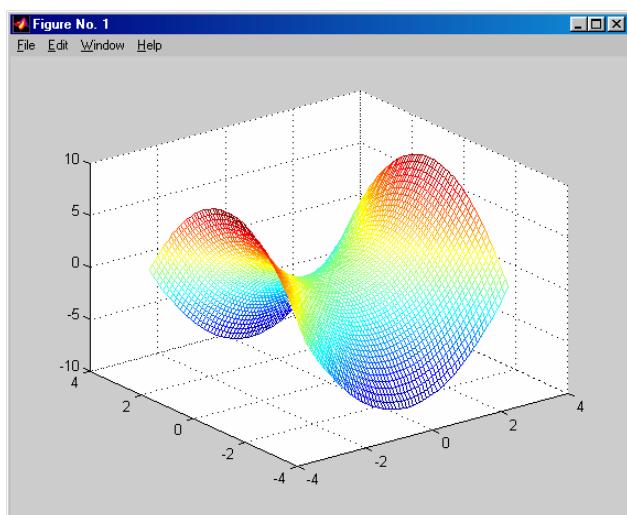
Niz naredbi MATLAB-a može se napisati u, na primer, MS-DOS Editoru i snimiti u datoteku za kasnije korišćenje. Ova datoteka pripada grupi tzv. skript datoteka ili „m-datoteke”. Možda često radite sa istim skupom podataka i tada je zgodno da napišete jednu m-datoteku kako ne biste stalno kucali iste naredbe u MATLAB-u. Kao primer, u MS-DOS Editoru napravićemo datoteku `skica.m`, koja ima sledeći sadržaj:

```
[x, y] = meshgrid(-3:.1:3, -3:.1:3);  
z = x.^2 - y.^2;  
mesh(x, y, z);
```

Kada pokrenemo MATLAB, pređite u svoj radni direktorijum, u ovom slučaju `dragan`, i u komandnoj liniji napišite naziv m-datoteke (`skica`).



MATLAB daje grafikon navedene funkcije.



Dobijen je rezultat kao da ste sva tri reda upisali direktno u MATLAB-u.

## Programiranje u MATLAB-u

Vrlo brzo ćete ustanoviti da možete mnoge probleme rešiti u MATLAB-u, koje ste do nedavno rešavali programiranjem u nekom višem programskom jeziku. Programi se pišu u nekom ASCI editoru (MS-DOS Editor) i snimaju kao m-datoteke. Važno je da zapamtite sledeće:

- Program se uređuje i snima kao i svaka druga datoteka.
- Program se snima u datoteku koja mora imati oznaku tipa .m.
- Program se izvršava u komandnom prozoru MATALAB-a.
- Kada hoćete da izvršite program u MATLAB-u samo navodite naziv datoteke (bez sufiksa .m).

Poželjno je da navodite komentare kod programiranja svojih funkcija, kako bi se kasnije lakše snašli ako treba nešto da prepravljate.

Pogodno je da sve svoje programe (datoteke) smestite u jedan direktorijum i kada želite da ih pozovete, neće vam biti problem da se setite gde se nalaze.

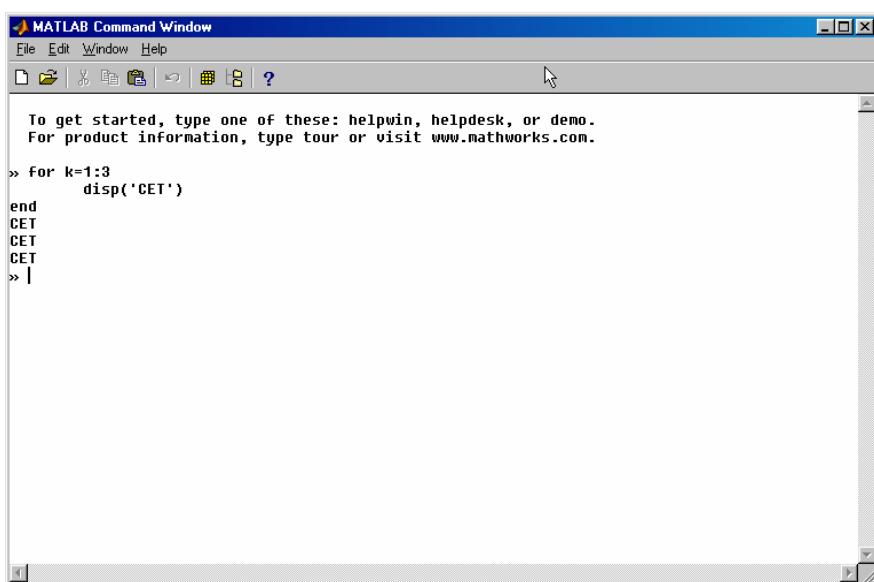
### Petlje **for** i **while**

MATLAB ima nekoliko procedura za kontrolu toka izvršavanja programa. One obuhvataju korišćenje naredbi **for**, **while** i **if**.

Petlja **for** ima sledeći oblik:

```
for k = 1:n
    ...
    (naredbe programa)
    ...
end
```

Naredbe programa između **for** i **end** ponavljaće se n puta. Kao primer dajemo sledeću sesiju u MATLAB-u:



The screenshot shows the MATLAB Command Window interface. The menu bar includes File, Edit, Window, and Help. The toolbar contains icons for New, Open, Save, Print, Copy, Paste, Find, and Help. The main window displays the following text:

```
To get started, type one of these: helpwin, helpdesk, or demo.
For product information, type tour or visit www.mathworks.com.

>> for k=1:3
       disp('CET')
end
CET
CET
CET
>> |
```

Petlja **while** omogućava da ponavljate niz naredbi sve dok je isti logički uslov zadovoljen. Ona ima sledeći oblik:

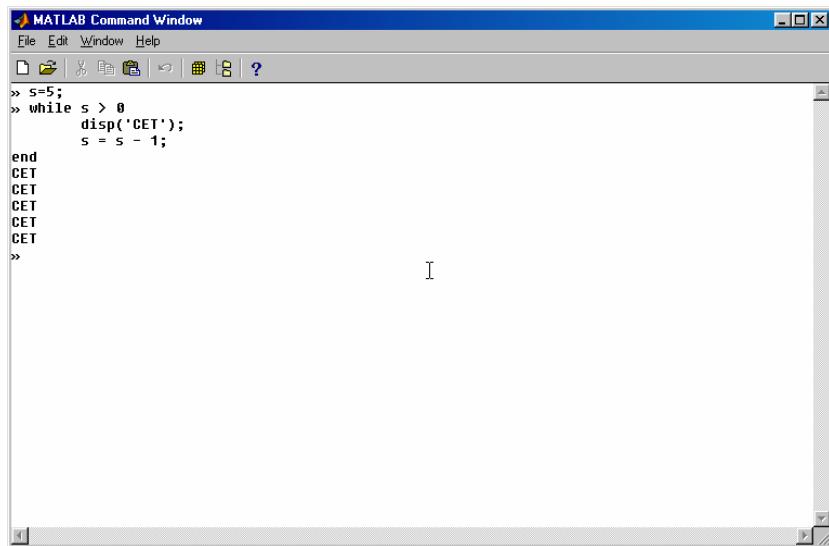
```

while uslov
    ...
    (naredbe programa)
    ...

end

```

Evo primera i za to:



### Testiranje pomoću **if** naredbe

Uobičajeni oblik if naredbe je:

```

if (uslov)
    ...
    (naredbe programa)
    ...
end

```

Ako je **uslov** ispunjen izvršava se niz naredbi između **if** i **end**, a ako nije taj niz se preskače.

### Funkcije

Funkcije su slične m-datotekama. Međutim, one prihvataju argumente i MATLAB prevodi (kompajlira) funkcije prilikom prvog izvršavanja u datoru radnoj sesiji, tako da se ubrzavaju potonja izvršavanja te funkcije.

Sada ćemo u MS-DOS Editoru napraviti datoteku **koren.m** koja sadrži sledeće redove:

```

function koren(x)
% Izracunava kvadratni koren Njutnovom metodom

% Pocetna vrednost
xstart = 1;

```

```

for i = 1:100
    xnovo = (xstart + x/xstart) / 2;
    disp(xnovo);
    if abs(xnovo - xstart) / xnovo < eps, break, end;
    xstart = xnovo;
end

```

### Napomene:

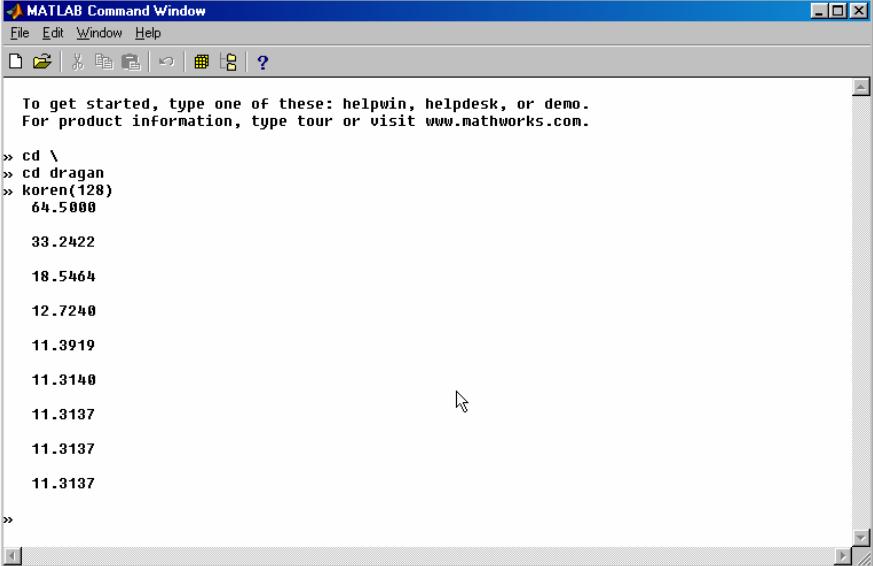
Znak % na početku reda označava da se radi o komentaru.

MATLAB ima definisanu konstantu `eps` koju koristi za toleranciju kod izračunavanja.

U MATLAB-u pređite u svoj radni direktorijum, u ovom slučaju `dragan`, i navedite u komandnoj liniji naziv funkcije i vrednost argumenta.

`koren(128)`

Ova funkcija će dati odziv koji je prikazan na sledećoj slici.



The screenshot shows the MATLAB Command Window. The title bar says "MATLAB Command Window". The menu bar includes "File", "Edit", "Window", and "Help". Below the menu is a toolbar with icons for file operations like Open, Save, and Help. The command window itself contains the following text:

```

To get started, type one of these: helpwin, helpdesk, or demo.
For product information, type tour or visit www.mathworks.com.

>> cd \
>> cd dragan
>> koren(128)
 64.5000

 33.2422
 18.5464
 12.7248
 11.3919
 11.3148
 11.3137
 11.3137
 11.3137

>>

```

### Napomene:

- Funkcija ima pristup promenljivama radnog okruženja iz koga se poziva. Međutim, promenljive koje se stvaraju unutar funkcije (u ovom slučaju, `xstart` i `xnovo`) su lokalne, što znači da se one ne mogu deliti u okviru radnog okruženja.
- Ukoliko prepravljate funkciju u toku svoje radne sesije, koristite naredbu `clear ime_funkcije` da biste uklonili prevedenu verziju funkcije iz radnog okruženja i omogućili da se učita nova verzija.

### Zaglavlje MATLAB funkcije

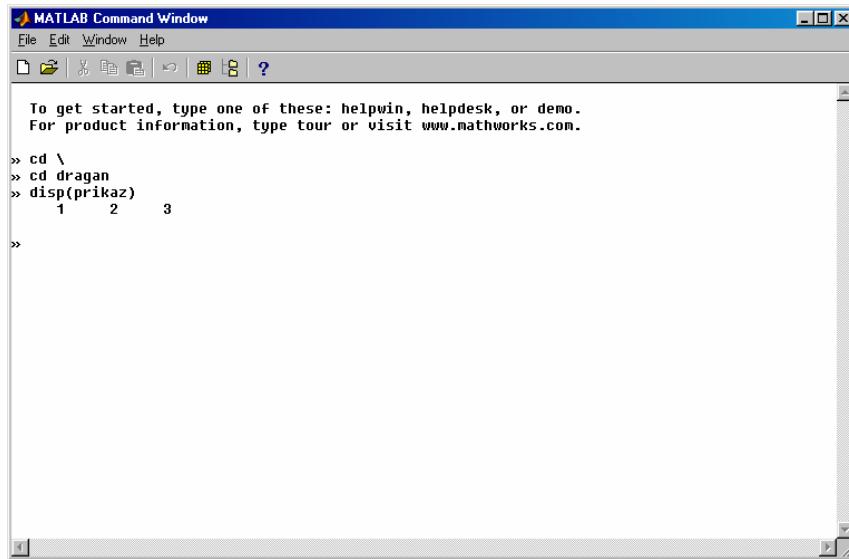
Informacije koje se pojavljuju u zaglavlu MATLAB funkcije posle reči `function` imaju sledeće značenje:

- Koji niz(nizove) funkcija vraća, ako uopšte vraća.
- Koji je naziv funkcije.
- Koji argumenti se prenose funkciji, ako ima argumente.

Nekoliko vrednosti može se dodeliti funkciji i ona može vratiti više vrednosti, što ćete videti na sledećem primeru funkcije prikaz.

```
function [a,b,c] = prikaz
% funkcija [a,b,c]=prikaz demonstr. sta sve funkcija moze da vrati
a=[1 2 3];
b='xyz';
c=[1 2 3; 4 5 6];
```

Možemo da vratimo nizove koje želimo. Ako navedemo ovu funkciju bez navođenja koji niz(nizove) želimo da vrati, ona će vratiti samo prvi.

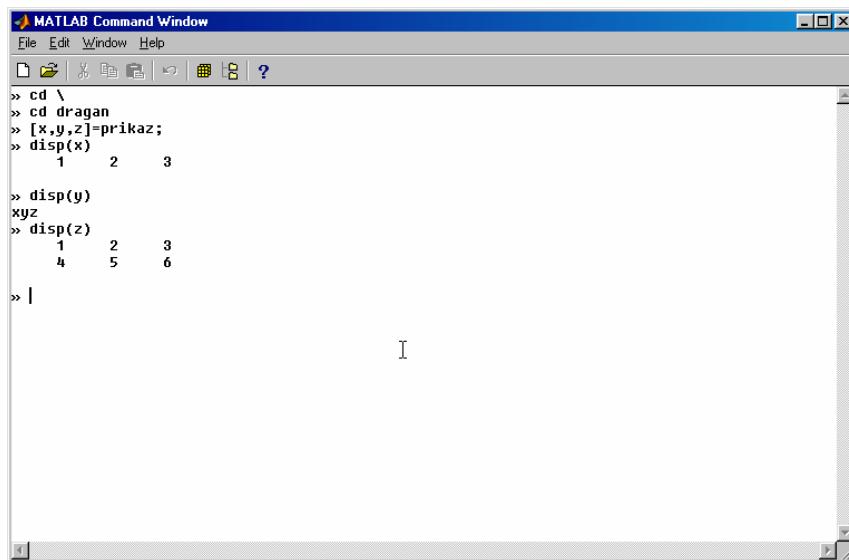


```
To get started, type one of these: helpwin, helpdesk, or demo.
For product information, type tour or visit www.mathworks.com.

>> cd \
>> cd dragan
>> disp(prikaz)
    1     2     3

>>
```

Ako navedemo više nizova, dobićemo tri koja su navedena u zaglavlju funkcije:



```
To get started, type one of these: helpwin, helpdesk, or demo.
For product information, type tour or visit www.mathworks.com.

>> cd \
>> cd dragan
>> [x,y,z]=prikaz;
>> disp(x)
    1     2     3

>> disp(y)
xyz
>> disp(z)
    1     2     3
    4     5     6

>> |
```

Vraćene promenljive ne moraju da budu istog oblika ili tipa, što se vidi iz prethodne sesije.

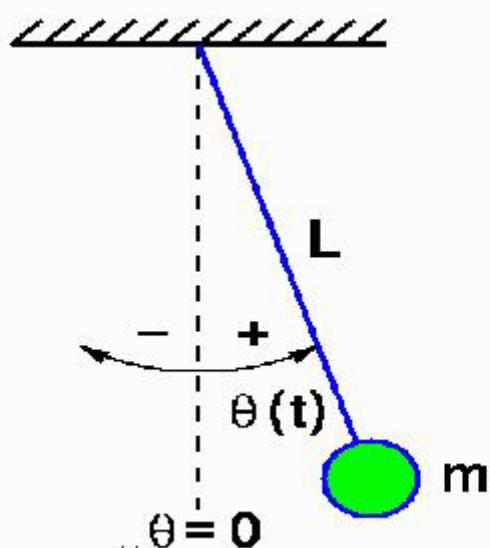
**Napomena:** Jedno od svojstava MATLAB programa je da mogu da koriste isti program sa promenljivim brojem argumenata. Promenljiva nargin može da se koristi unutar

bilo kog funkcionskog programa za pronalaženje broja argumenata koji se koriste u sesiji koja poziva funkciju.

## MATLAB (7)

### Primer rešavanja jednačine klatna

Razmotrimo klatno koje ima polugu dužine  $L$  i kuglu mase  $m$ .



Prepostavimo da je poluga kruta i da ima zanemarljivu mase u poređenju sa kuglom. Takođe, prepostavimo da nema trenja ili otpora vazduha. Koristeći II Njutnov zakon dobijamo sledeću diferencijalnu jednačinu za ugaoni pomeraj,  $\theta(t)$ :

$$\theta''(t) + (g/L)\sin(\theta(t)) = 0$$

Prepostavljajući da je kugla u startu pomerena za ugao  $\theta_0$  i da je puštena da se njiše, dobijamo sledeće početne uslove:

$$\theta(0) = \theta_0, \quad \theta'(0) = 0.$$

Diferencijalnu jednačinu sa ovim početnim uslovima rešavamo koristeći Runge Kutta Fehlberg metod 4-og reda, tj. funkciju ODE45.

Ukoliko u komandnoj liniji MATLAB-a navedete

```
help ode45
```

dobićete iscrpno objašnjenje.

Za nas je u ovom slučaju od interesa da objasnimo sledeći oblik funkcije ODE45:

```
[T, Y] = ODE45 ('F', TSPAN, Y0),
```

gde je TSPAN = [TO TFINAL] interval na kome se integriše sistem diferencijalnih jednačina  $y' = F(t, y)$ , uz početni uslov  $Y0$ . 'F' je string koji sadrži naziv ODE datoteke. Funkcija  $F(T, Y)$  mora da vrati kolona vektor. Svaka vrsta u rešenju  $Y$  odgovara

vremenu vraćenom u kolona vektoru T. Da biste dobili rešenje za određene trenutke T0, T1, ..., TFINAL, koristite TSPAN = [T0 T1 ... TFINAL].

Oblik

[T, Y] = ODE45 ('F', TSPAN, Y0, OPTIONS)

radi na sličan način, samo su parametri integracije definisani preko parametra OPTIONS, koji daje funkcija ODESET. (Za detaljnije objašnjenje u komandnoj liniji treba navesti help odeset.) Obično se koriste skalarna vrednost, tolerancija relativne greške 'RelTol' (podrazumeva se vrednost 1e-3), i vektorska vrednost, tolerancija apsolutne greške 'AbsTol' (podrazumeva se vrednost 1e-6 za sve komponente).

Vratimo se sada našem problemu. Prvo ćemo definisati ekvivalentni sistem diferencijalnih jednačina 1-og reda. Definišimo vektore  $u_1 = \theta$ , i  $u_2 = \theta'$ . Sada imamo sledeći sistem:

$$u_1'(t) = u_2(t), \quad u_1(0) = \theta_0$$

$$u_2'(t) = -g^2 \sin(u_1(t)), \quad u_2(0) = 0$$

gde je  $g^2 = g/L$ .

Definisaćemo sada m-datoteku za prosto klatno, tj. klatnou.m

```
function uprim=klatnou(t,u)
% jednacina prostog klatna
% uprim(1)=u(2)
% uprim(2)=-omega*sin(u(1))
omega=16;uprim=[u(2);-omega*sin(u(1))];
```

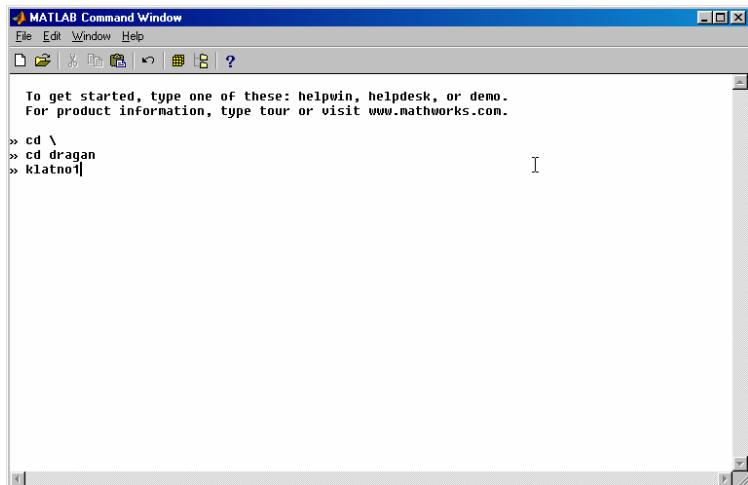
Definisaćemo sada m-datoteku za linearizovan model (za male uglove možemo smatrati da je  $\sin \theta = \theta$ ), tj. klatnow.m

```
function wprim=klatnow(t,w)
% linearizovana jednacina klatna
omega=16;wprim=[w(2);-omega*w(1)];
```

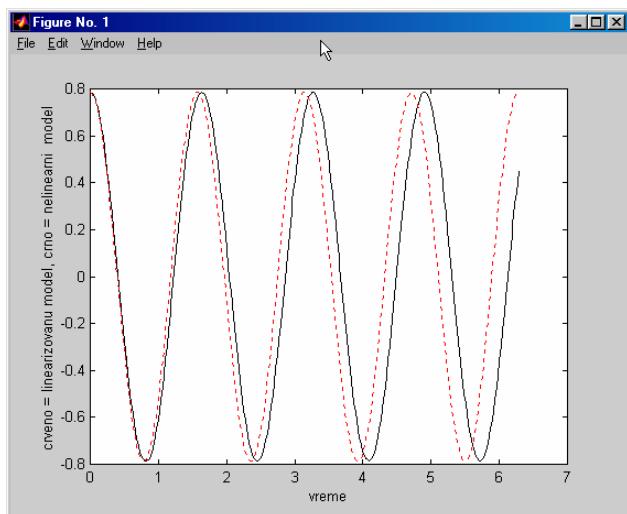
„Glavni” program je m-datoteka klatno1.m koja poziva prethodne funkcije, tj. m-datoteke.

```
% model prostog klatna
clear all,tspan=[0,2*pi];u0=[pi/4;0];
opcije=odeset('RelTol',1e-5,'AbsTol',1e-8)
[t,u]=ode45('klatnou',tspan,u0,opcije);
% za linearizovan model prostog klatna
[s,w]=ode45('klatnow',tspan,u0,opcije);
plot(t,u(:,1),'k-',s,w(:,1),'r:')
xlabel('vreme')
ylabel('crveno = linearizovanu model, crno = nelinearni
model')
```

Pokrenimo MATLAB i proverimo kako ovo funkcioniše.



The screenshot shows the MATLAB Command Window. It displays the standard startup message: "To get started, type one of these: helpwin, helpdesk, or demo. For product information, type tour or visit www.mathworks.com." Below this, the user has entered the command: `>> cd \`, `>> cd dragan`, and `>> klatno1`. The cursor is positioned at the end of the last command.



Pripremio Dragan Marković