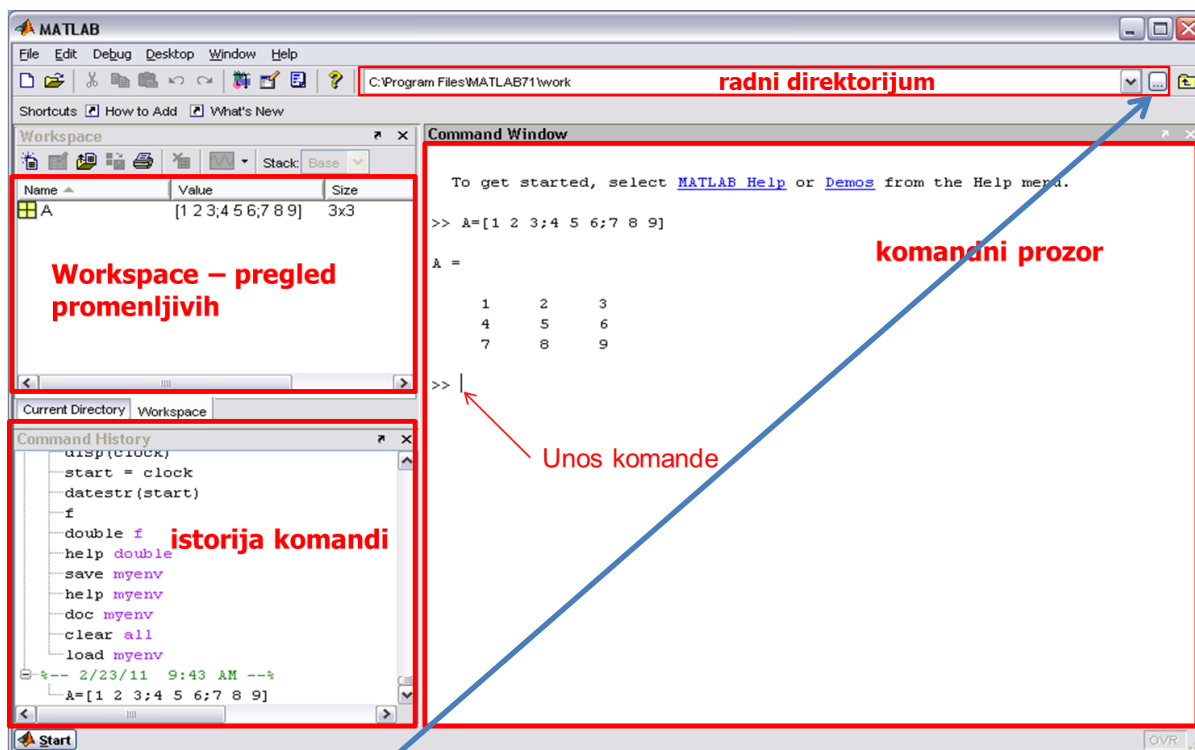


# 1. Uvod u MATLAB

MATLAB spada u grupu programa zasnovanih na *numeričkom* rešavanju matematičkih problema. Predstavlja interpreter zasnovan na matičnom računu i koncipiran je kao proširiv programski paket (neki od toolbox-ova koje Matlab sadrži su *ControlSystem Toolbox*, *SignalProcessing Toolbox* i *NeuralNetwork Toolbox*).

Osnovno okruženje predstavlja tekstualni prozor u kojem se unose *matlab* komande ili naredbe. Komande se izvršavaju neposredno nakon unošenja. Poseban simbol „>>” predstavlja početak komande. Pored komandi se mogu pisati i programi u obliku funkcija ili procedura.

Na slici 1 je predstavljeno MATLAB okruženje i najvažnije komponente u okviru njega (radni direktorijum – u okviru ovog direktorijuma se traže .m datoteke sa korisnički definisanim funkcijama; *workspace* – služi za pregled deklarisanih promenljivih i njihovih vrednosti; istorija komandi – pregled komandi koje je korisnik izvršio; komandni prozor – služi za unos novih komandi).



Slika 1. Matlab okruženje

**Podešavanje radnog direktorijuma:** U direktorijumu koji koristite za vežbe kreirati direktorijum vezba1 i podesiti radni direktorijum matlab okruženja na kreirani direktorijum.

## Definisanje promenljivih

Osnovni element u Matlabu je matrica kompleksnih brojeva. Dva moguća načina **definisanja matrice**  $A$  (u ovom slučaju dimenzija  $3 \times 3$ ) su:

```
>> A = [1 2 3; 4 5 6; 7 8 9]
```

A =

1	2	3
4	5	6
7	8	9

ili

```
>> A = [  
1 2 3  
4 5 6  
7 8 9]
```

A =

1	2	3
4	5	6
7	8	9

Primer definisanja **vektora-vrste** je:

```
>> x = [-1.3 sqrt(3.1) (1+2)/4*5]
```

x =

-1.3000	1.7607	3.7500
---------	--------	--------

Definicija **imagirne jedinice**:

```
>> i = sqrt(-1)
```

i =

0.0000 + 1.0000i

Definisana promenljiva se može koristiti u novim naredbama, npr.

```
>> C = [1+5*i 2+6*i; 3+7*i 8]
```

C =

1.0000 + 5.0000i	2.0000 + 6.0000i
3.0000 + 7.0000i	8.0000 + 0.0000i

**Uređeni niz brojeva** se može definisati na sledeći način:

```
>> x = 1:4
```

x =

1	2	3	4
---	---	---	---

, a za korak različit od 1:

```
>> y = 1:0.5:3
```

```
y =
```

```
1.0000    1.5000    2.0000    2.5000    3.0000
```

Operator : može se koristiti i na sledeći način:

```
>> A = [1 2 3; 4 5 6; 7 8 9];
```

```
>> A(:) = 21:29
```

```
A =
```

```
21    24    27
22    25    28
23    26    29
```

## Selekcija elemenata matrice

Za matricu:

```
>> A = [1 2 3; 4 5 6; 7 8 9]
```

```
A =
```

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

I. Element A[2][2] se dobija na sledeći način:

```
>> A(2,2)
```

```
ans =
```

```
5
```

II. Izdvajanje 3. kolone matrice A:

```
>> A(:,3)
```

```
ans =
```

```
3
6
9
```

III. Podmatrica:

```
>> A([1 3], [2 3])
```

```
ans =
```

```
2    3
8    9
```

#### IV. Proširenje matrice:

```
>> A = [A; 10 11 12]
```

```
A =
```

```
     1     2     3
     4     5     6
     7     8     9
    10    11    12
```

#### V. Uklanjanje 2. kolone:

```
>> A = A(:, [1 3])
```

```
A =
```

```
     1     3
     4     6
     7     9
    10    12
```

#### VI. Elementi koji nisu eksplicitno definisani:

```
>> x = 1 : 4;
```

```
>> x(6) = 7
```

```
x =
```

```
     1     2     3     4     0     7
```

### Osnovne operacije sa matricama

I. **Transponovanje:**  $B = A'$

II. **Sabiranje:**  $C = A + B$

III. **Sabiranje matrice i skalara:**  $C = C - 2$

IV. **Množenje matrica:**  $D = A * B$

V. **Operator \** (za zadatu matricu  $A$  i vektor  $b$ ,  $A \backslash b$  predstavlja rešenje sistema  $Ax = b$ ):

$$\begin{aligned} 10x + 2y + 6z &= 28 \\ x + 10y + 9z &= 7 \\ 2x - 7y - 10z &= -17 \end{aligned}$$

$$A = \begin{bmatrix} 10 & 2 & 6 \\ 1 & 10 & 9 \\ 2 & -7 & -10 \end{bmatrix} b = \begin{bmatrix} 28 \\ 7 \\ -17 \end{bmatrix}$$

```
>> A = [10 2 6; 1 10 9; 2 -7 -10];
```

```
>> b = [28 7 -17]';
```

```
>> x = A\b
```

```
x =
```

```
     1.0000
    -3.0000
     4.0000
```

VI. **Pair-wise operacije sa elementima matrice (ili vektora):** `.*`, `.+`, `.-`, `./`, itd.:

```
>> x = [1 2 3];  
>> y = [4 5 6];  
>> x .* y  
  
ans =  
  
4    10    18
```

VII. **Stepenovanje:** `^`

```
>> A = [1 2 3; 4 5 6; 7 8 9];  
>> A ^ 2  
  
ans =  
  
30    36    42  
66    81    96  
102   126   150  
  
>> A .^ 2  
  
ans =  
  
1     4     9  
16    25    36  
49    64    81
```

VIII. **Relacioni operatori:**

```
manje: <  
manje ili jednako: <=  
veće: >  
veće ili jednako: >=  
jednako: ==  
različito: ~=  
  
>> 2 + 2 == 4  
  
ans =  
  
1  
  
>> A = [1 2 3; 4 5 6; 7 8 9];  
>> B = A < 5  
  
B =  
  
1     1     1  
1     0     0  
0     0     0
```

## IX. Logički operatori:

negacija: ~  
disjunkcija: |  
konjukcija: &

```
>> ~B
```

```
ans =
```

```
0    0    0
0    1    1
1    1    1
```

## Ugrađene funkcije:

kosinus:	<code>cos(x)</code>
sinus:	<code>sin(x)</code>
tangens:	<code>tan(x)</code>
$e^x$ :	<code>exp(x)</code>
$\sqrt{x}$ :	<code>sqrt(x)</code>
logaritam sa bazom $e$ :	<code>log(x)</code>
logaritam sa bazom 10:	<code>log10(x)</code>
apsolutna vrednost:	<code>abs(x)</code>
maksimum elemenata vektora ili matrice:	<code>max(x)</code>
minimum elemenata vektora ili matrice:	<code>min(x)</code>
suma elemenata vektora ili matrice:	<code>sum(x)</code>
zaokruživanje ka $+\infty$ :	<code>ceil(x)</code>
zaokruživanje ka $-\infty$ :	<code>floor(x)</code>
ostatak deljenja $x$ sa $y$ :	<code>rem(x, y)</code>

### I. Osnovni podaci o funkciji:

```
>> help sin
sin      Sine of argument in radians.
      sin(X) is the sine of the elements of X.

See also asin, sind.

Reference page for sin
Other functions named sin
```

### II. Detaljna dokumentacija vezana za funkciju:

```
>> doc sin
>>
```

### III. Pronalaženje svih MATLAB funkcija koje u dokumentaciji sadrže određenu reč:

```
>> lookfor logarithm
```

```
log
```

```
log10
```

```
log2
```

```
number.
```

```
reallog
```

```
logspace
```

```
logm
```

```
betaln
```

```
gammaln
```

- Natural logarithm.

- Common (base 10) logarithm.

- Base 2 logarithm and dissect floating point

- Real logarithm.

- Logarithmically spaced vector.

- Matrix logarithm.

- Logarithm of beta function.

- Logarithm of gamma function.

#### primer 1:

```
>> a = [1 2 -3 0; 4 0 -8 0]
```

```
a =
```

```
     1     2     -3     0
     4     0     -8     0
```

```
>> all(a)
```

```
ans =
```

```
     1     0     1     0
```

```
>> any(a)
```

```
ans =
```

```
     1     1     1     0
```

#### primer 2:

```
>> a = 5; x = 2; y = 8;
```

```
>> y = exp(-a)*sin(x) + 10*sqrt(y)
```

```
y =
```

```
28.2904
```

#### primer 3:

```
>> t = (0:0.2:1)';
```

```
>> y = sin(t) .* exp(-t);
```

```
>> [t y]
```

```
ans =
```

```
     0     0
0.2000 0.1627
0.4000 0.2610
0.6000 0.3099
0.8000 0.3223
1.0000 0.3096
```

## MATLAB konstante:

$\pi$ : pi  
 $\infty$ : Inf  
nedefinisana vrednost: NaN

### primer 1:

```
>> pi  
  
ans =  
  
3.1416  
  
>> sin(pi/4)  
  
ans =  
  
0.7071
```

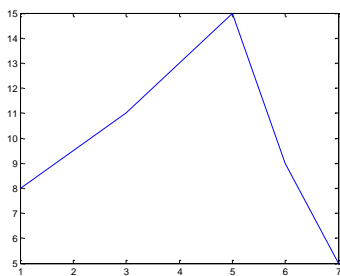
### primer 2:

```
>> 0/0  
  
ans =  
  
NaN  
  
>> 1/0  
  
ans =  
  
Inf
```

## Crtanje grafika funkcija

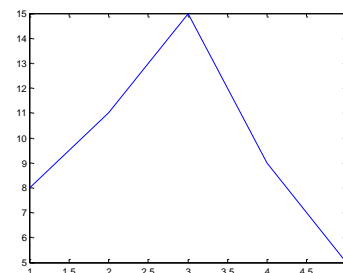
Funkcija `plot(x, y, ...)` crta grafik tačaka u 2D. Tačke se prosleđuju kao vektori koordinata:

```
>> x = [1 3 5 6 7];  
>> y = [8 11 15 9 5];  
>> plot(x,y)
```



Ako se ne prosledi vektor `x` koordinata već samo `y`, MATLAB će koristiti poziciju u vektoru `x` kao koordinatu:

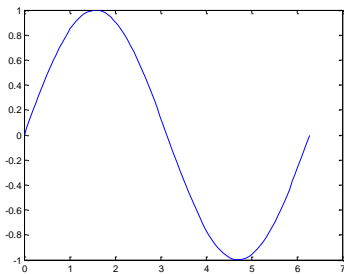
```
>> plot(y)
```





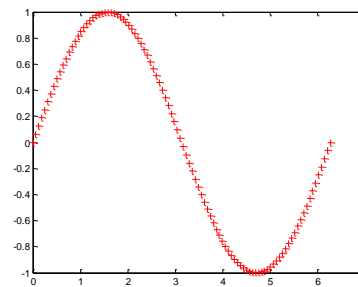
Crtanje grafika funkcije sinus na intervalu od 0 do  $2\pi$ , tako da na grafiku bude 100 tačaka:

```
>> x = linspace(0,2*pi,100);  
>> y = sin(x);  
>> plot(x,y)
```



Isti grafik ali sa linijom crvene boje, a tačke su prikazane znakom +:

```
>> plot(x,y,'r+')
```



### Dodavanje naslova i oznaka osa.

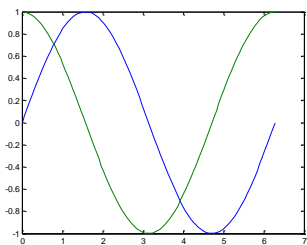
Dok je grafik otvoren ukucati:

```
>> title('grafik sinus funkcije')  
>> xlabel('x osa')  
>> ylabel('y osa')
```

### Crtanje više grafika na istoj slici

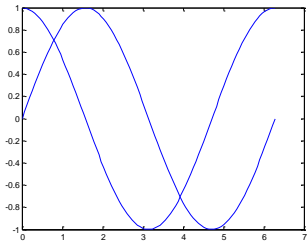
Prvi način je prosleđivanje vektora koordinata za oba grafika:

```
>> x1 = linspace(0,2*pi,100);  
>> y1 = sin(x);  
>> x2 = linspace(0,2*pi,100);  
>> y2 = cos(x);  
>> plot(x1,y1,x2,y2)
```



Drugi način je crtanje jednog grafika pa dodavanje novog na postojeću sliku:

```
>> plot(x1,y1,x2,y2)
>> plot(x1,y1), hold
Current plot held
>> plot(x2,y2)
```



## Workspace

### 1. Snimanje istorije komandi

Istorija komandi se može snimati u tekstualnu datoteku:

```
>> diary mojekomande
>> a = [1 2 3]
>> b = [4 5 6]
>> diary off
```

Istoriju komandi možete pogledati ako otvorite datoteku „mojekomande“ u bilo kom tekstualnom editoru. Datoteka se nalazi u radnom direktorijumu.

### 2. Čuvanje promenljivih iz *workspace*-a

Promenljive iz *workspace*-a se mogu snimiti u datoteku uz pomoć komande:

```
>> save mejepromenljive
```

### 3. Učitavanje snimljenih promenljivih:

```
>> load mejepromenljive
```

### 4. Brisanje promenljivih iz *workspace*-a:

```
>> clear all
```

### 5. Brisanje komandnog prozora:

```
>> clc
```

## Programi

Programi (skripte) su skup MATLAB naredbi i smeštaju se u **.m** datoteke. Naredbe se izvršavaju jedna po jedna i bilo kakva greška prekida program. Sve što je do tada izračunato (promenljive), ostaje u *workspace*-u.

Skripte se pokreću po nazivu:

```
>> moj_program
```

### Funkcije:

Funkcije se takođe smeštaju u **.m** datoteke:

1. Tipično se piše jedna funkcija u jednoj .m datoteci. Ako to nije slučaj, prva funkcija u datoteci mora imati isti naziv kao i sama datoteka.
2. Da bi se funkcija mogla pozvati u skripti ili drugoj funkciji, ona mora biti ili predefinisana MATLAB funkcija ili se njena .m datoteka mora nalaziti u workspace-u iz koga je pozvana.

#### *naziv\_funkcije.m*

---

```
% komentar
function [povratna_vrednost1, povratna_vrednost2,...] = naziv_funkcije(parametar1, parametar2,...)
    .
    .
    .

    povratna_vrednost1 = ...; % vraćanje vrednosti
end
```

---

### Iteracija i selekcija:

<pre>for indeks_petlje = vektor     .     .     . end</pre>	<pre>while izraz     .     .     . end</pre>	<pre>if izraz1     .     .     . elseif izraz2 % opciono     .     .     . else % opciono     .     .     . end</pre>
---	--	---

### Kontrola toka:

<pre>return</pre>	<pre>break</pre>	<pre>continue</pre>
-------------------	------------------	---------------------

### Vraćanje vrednosti pre kraja funkcije:

```
function povratna_vrednost = naziv_funkcije(parametar1, parametar2,...)
    .
    .
    .
    if izraz
        povratna_vrednost = ...;
        return
    end
    .
    .
    .
end
```