RAČUNARSKI ALATI - MATLAB

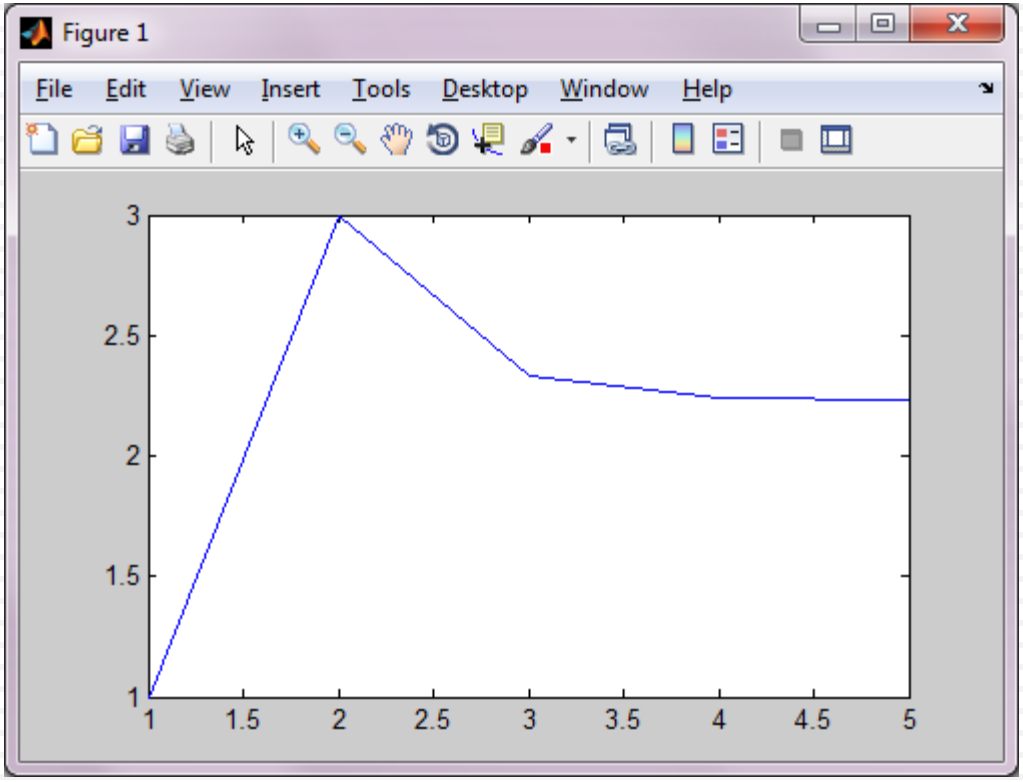
GRAFIKA

2D Grafika

- Formirajmo niz *x* aproksimacija vrednosti $\sqrt{5}$

```
>> x(1)=1;
>> for i=2:5
x(i)=(x(i-1)+5/x(i-1))/2;
end
```

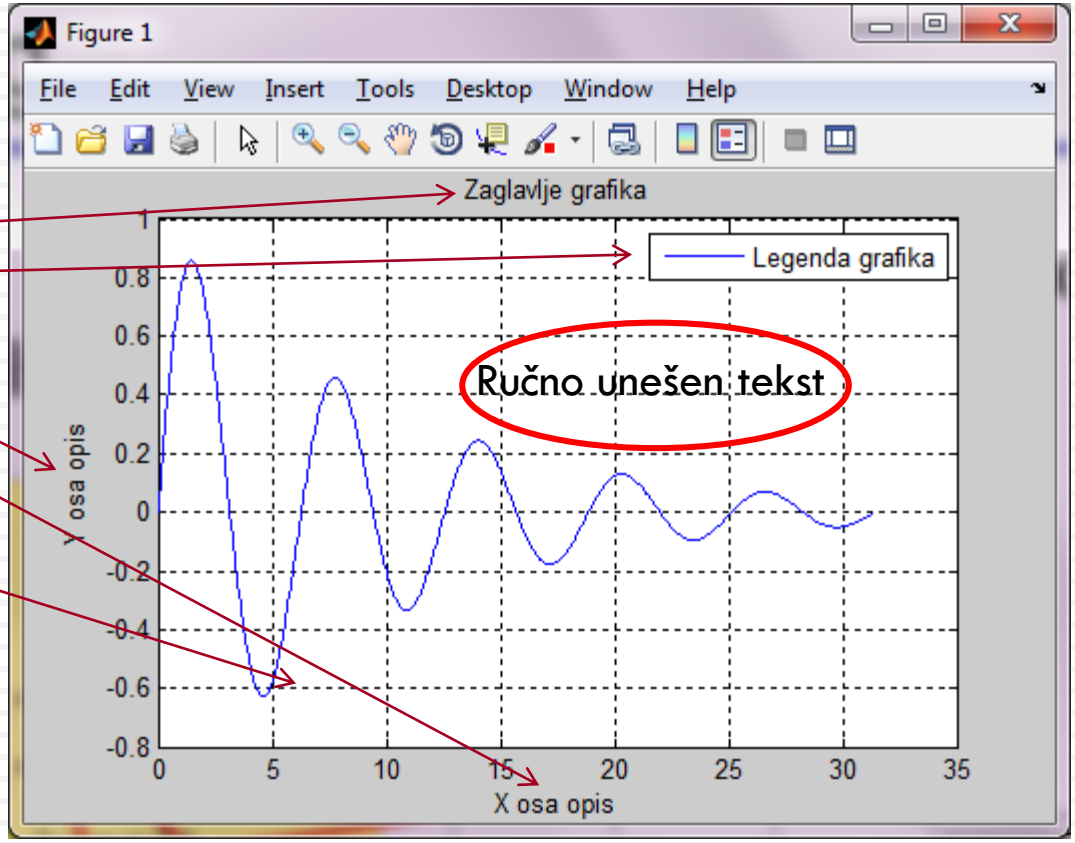
- plot(x) crta vektor x, povezujući linijom tačke (1,x(1)), (2,x(2)), …



2D Grafika

- plot(x,y) povezuje skup tačaka u ravni koje su zadane vektorima x odnosno y koordinata (moraju biti iste dužine)
  - Grafički prikazuje zavisnost y od x

```
>> x=0:0.05:10*pi;
>> y=exp(-.1.*x).*sin(x);
>> plot(x,y)
>> title('Zaglavlje grafika')
>> legend('Legenda grafika')
>> ylabel('Y osa opis')
>> xlabel('X osa opis')
>> grid on
```



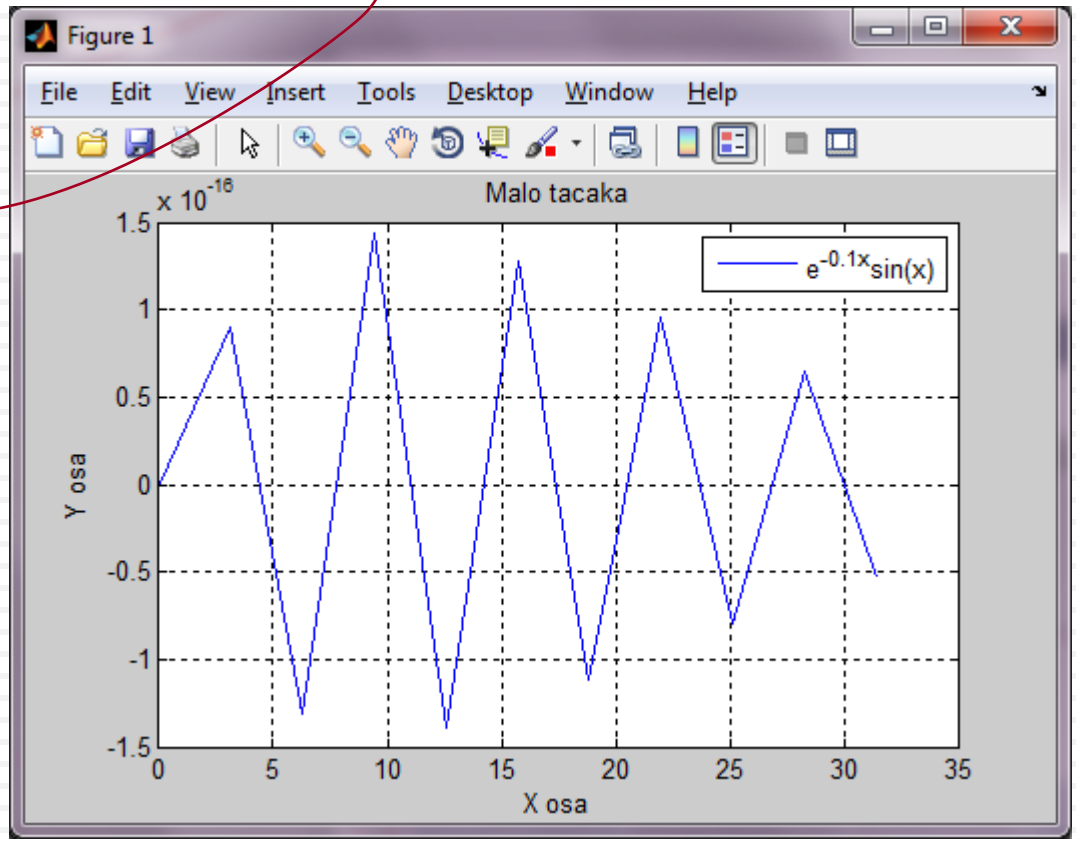
Obrtanje redosleda x, y rotira grafik za 90 stepeni

plot(x) ima isti efekat kao plot(1:length(x), x)

2D Grafika

- Ako tačke nisu dovoljno guste

```
>> x=0:pi:10*pi;
>> y=exp(-.1.*x).*sin(x);
>> plot(x,y)
>> title('Malo tacaka')
>> legend('e^{-0.1x}sin(x)')
>> ylabel('Y osa')
>> xlabel('X osa')
>> grid on
```



Dekoracija

- title - naslov dijagrama
- xlabel, ylabel - slovne oznake apscise i ordinate
- text, gtext - ispis teksta na proizvoljnom mestu
- grid on/off - prikaz mreže
- legend - prikaz legende

Stilovi linija, boje i markeri

- plot(x,y, 'sbm'): Karakteri *sbm* redom specificiraju stil linije, boju i marker (tačke). Ne moraju se svi navoditi

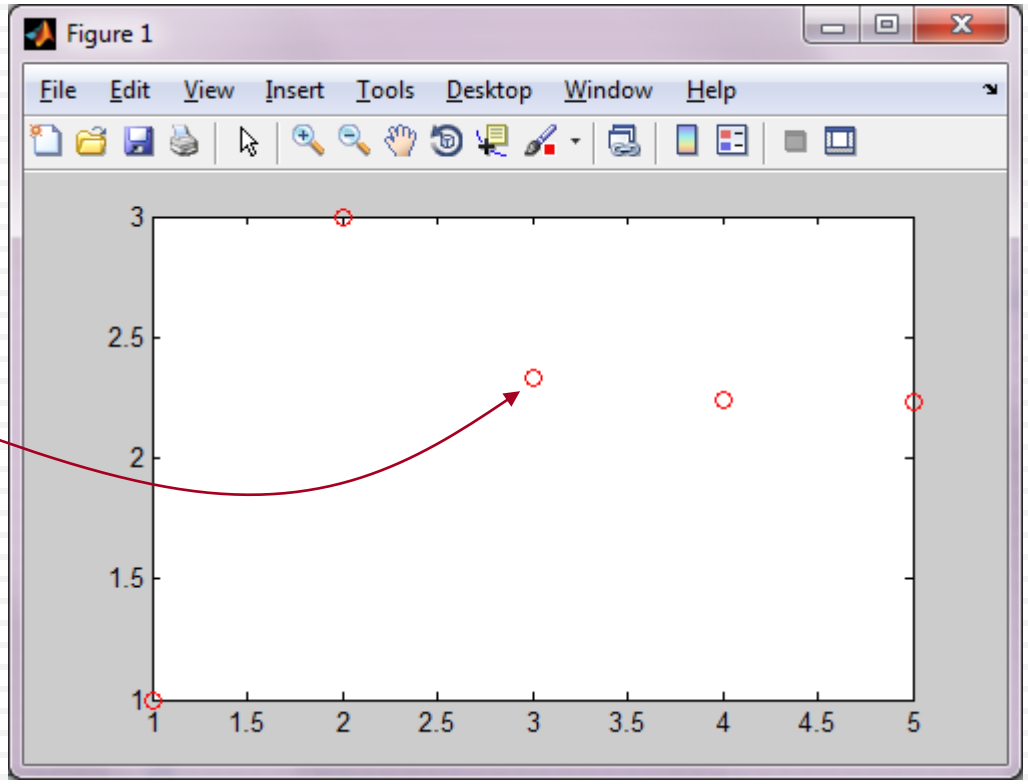
Stil linije	
-	puna (podrazumevani stil)
:	tačkasta
-.	crta tačka
--	crtice
	nema linije

Boja	
b	blue
g	green
r	red
c	cyan
m	magenta
y	yellow
k	black
w	white

Marker	
.	tačka
o	kružić
x	znak x
+	znak +
*	zvezdica
s	kvadrat
d	romb
v	trougao (na dole)
^	trougao (na gore)
<	trougao (na levo)
>	trougao (na desno)
p	petokraka zvezda
h	šestokraka zvezda

Stilovi linija, boje i markeri

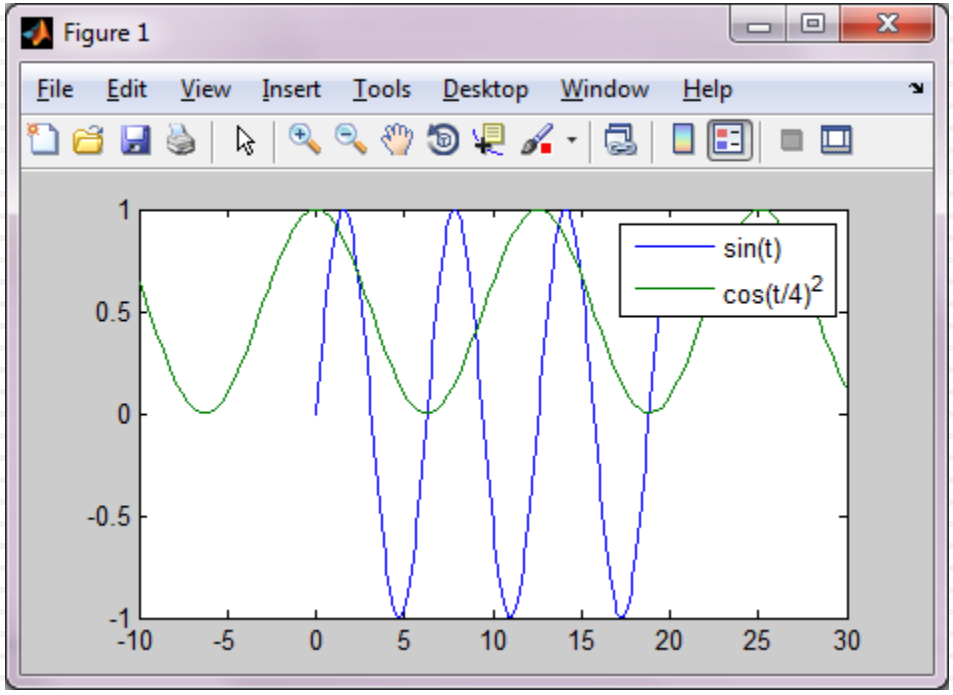
```
>> x(1)=1;
>> for i=2:5
x(i)=(x(i-1)+5/x(i-1))/2;
end
>> plot(x,'ro')
```



Crtanje više krivih

- Zadavanjem dvojki parametara (trojki, u slučaju dekoracije)
  - Krive ne moraju biti zadate u jednakom broju tačaka

```
>> x1=0:0.1:20;
>> y1=sin(x1);
>> x2=-10:0.25:30;
>> y2=cos(x2/4).^2;
>> plot(x1,y1,x2,y2.*y2);
>> legend('sin(t)',…
'cos(t/4)^2');
```

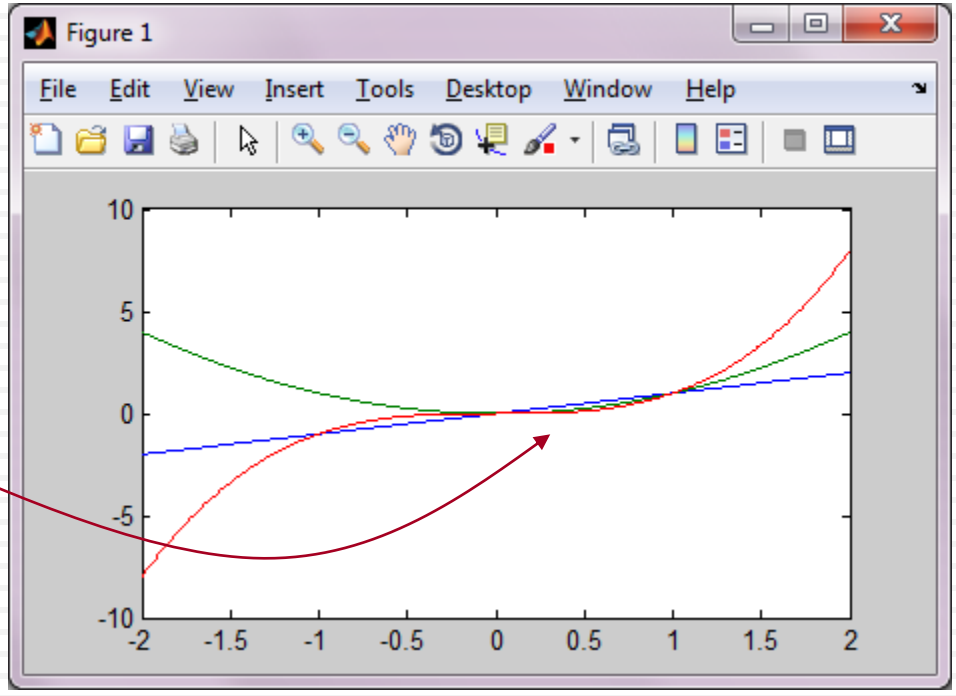


Crtanje više krivih

- Zadavanjem nekoliko krivih u kolononama matrice
- Krive moraju biti zadate u istim tačkama

```
>> x=(-2:0.01:2)';
>> plot(x,[x, x.*x,…
x.^3]);
```

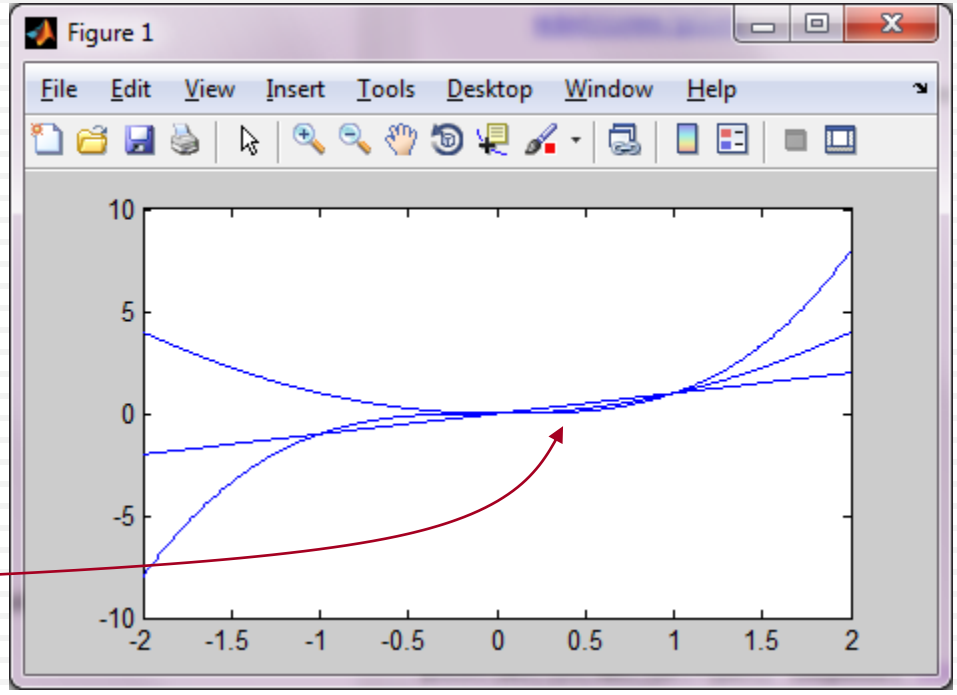
plot funkcija automatski svakoj sledećoj krivoj koju crta pridružuje sledeću boju iz spiska raspoloživih boja



Crtanje više krivih

- Naknadno dodavanje krivih (ose su "zamrznute")
  - hold on/off: Postojeći sadržaj grafičkog prozora se zadržava ili ne

```
>> x=(-2:0.01:2);
>> plot(x,x);
>> hold on;
>> plot(x,x.^2);
>> plot(x,x.^3);
>> hold off;
```

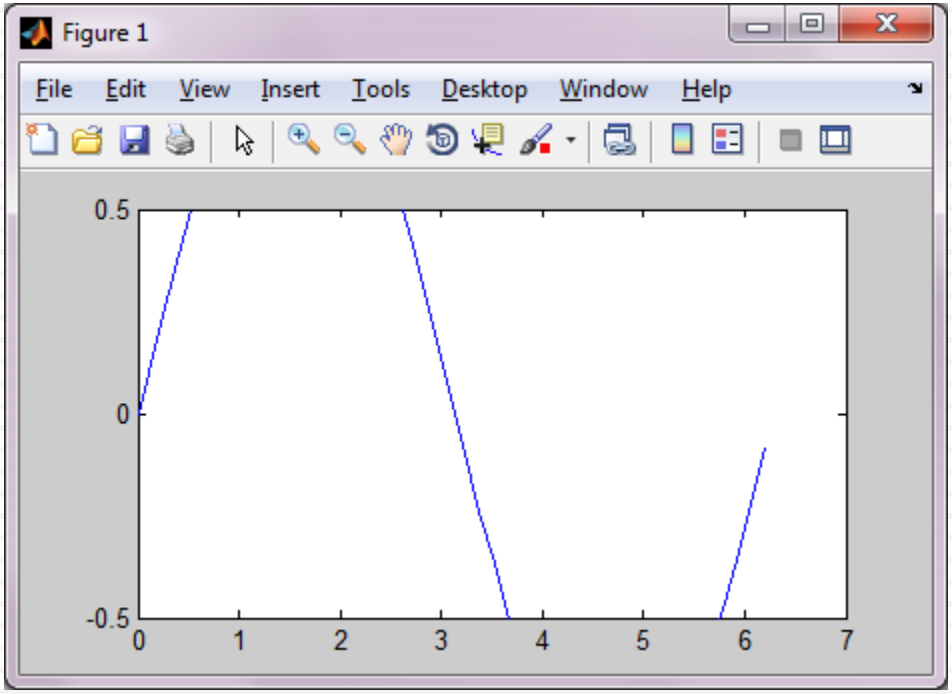


Svaki poziv plot funkcije koristi istu boju (ako se ne zada)

2D Grafika

- Prikaz crteža se može kontrolisati sledećom funkcijom
  - axis: Skaliranje osa

```
>> x=0:0.1:2*pi;
>> plot(x,sin(x))
>> axis
ans =
     0     7    -1     1
>> axis([0 7 -.5 .5])
```



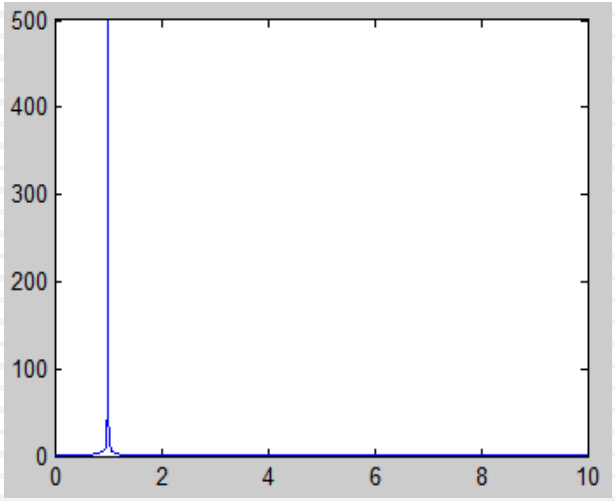
Funkcije za crtanje 2D grafikona

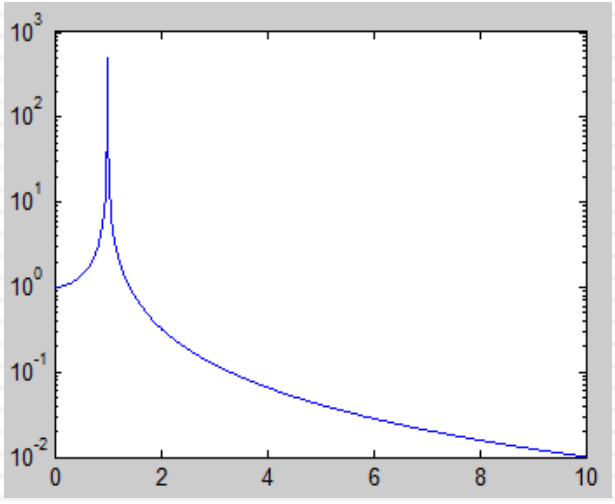
- Skaliranja osa je automatsko
- Mogu se birati skale za ose
  - plot - crtanje dijagrama sa linearnom podelom na obe ose
  - loglog - logaritamska podela na obe ose
  - semilogx - apscisa logaritamska, ordinata linearna
  - semilogy - apscisa linearna, ordinata logaritamska

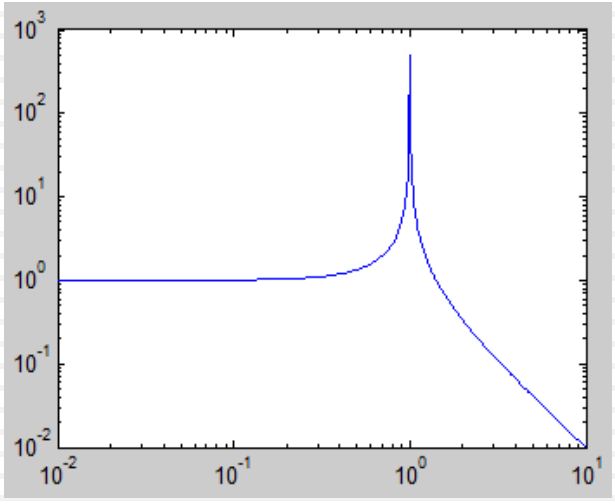
Logaritamske skale

- U slučaju velikih promena vrednosti koristi se logaritamska skala

```
>> x=linspace(0,10,1000);
>> y=1./abs(1-x.^2);
>> plot(x,y);
>> semilogx(x,y)
>> loglog(x,y)
```

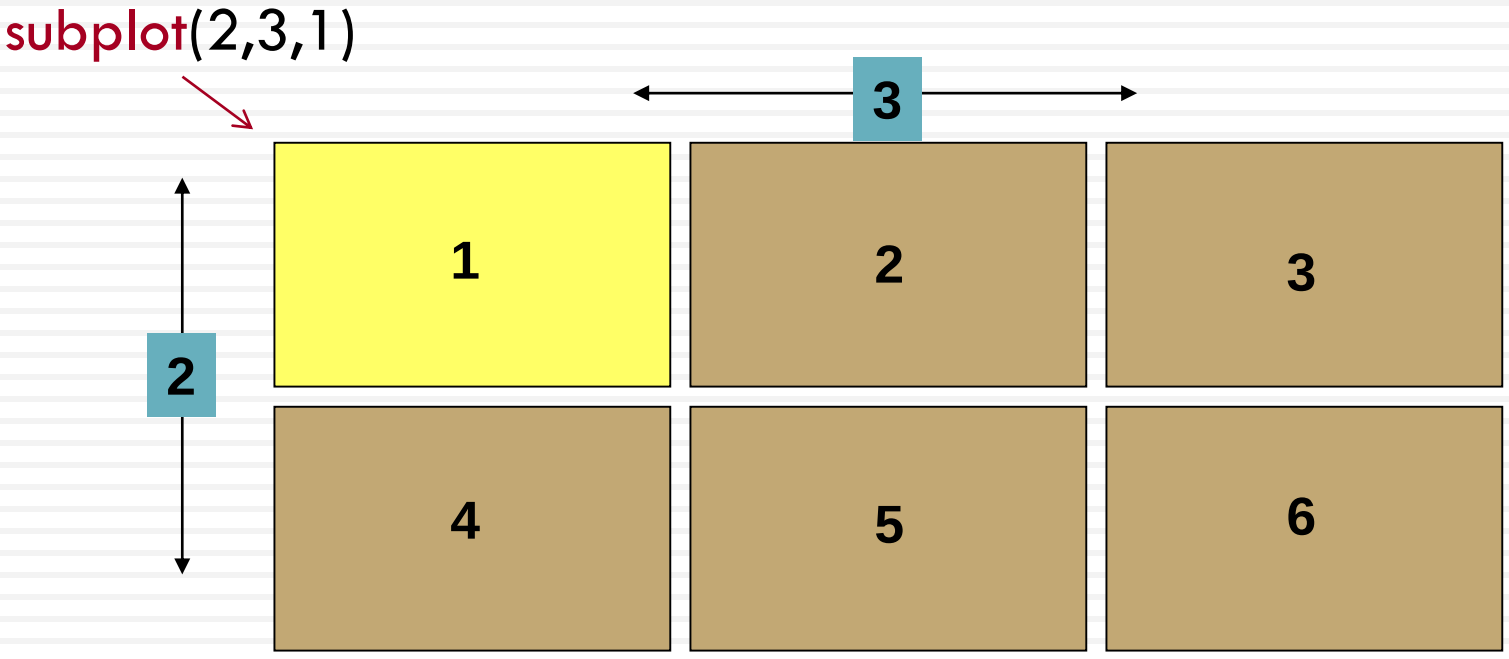






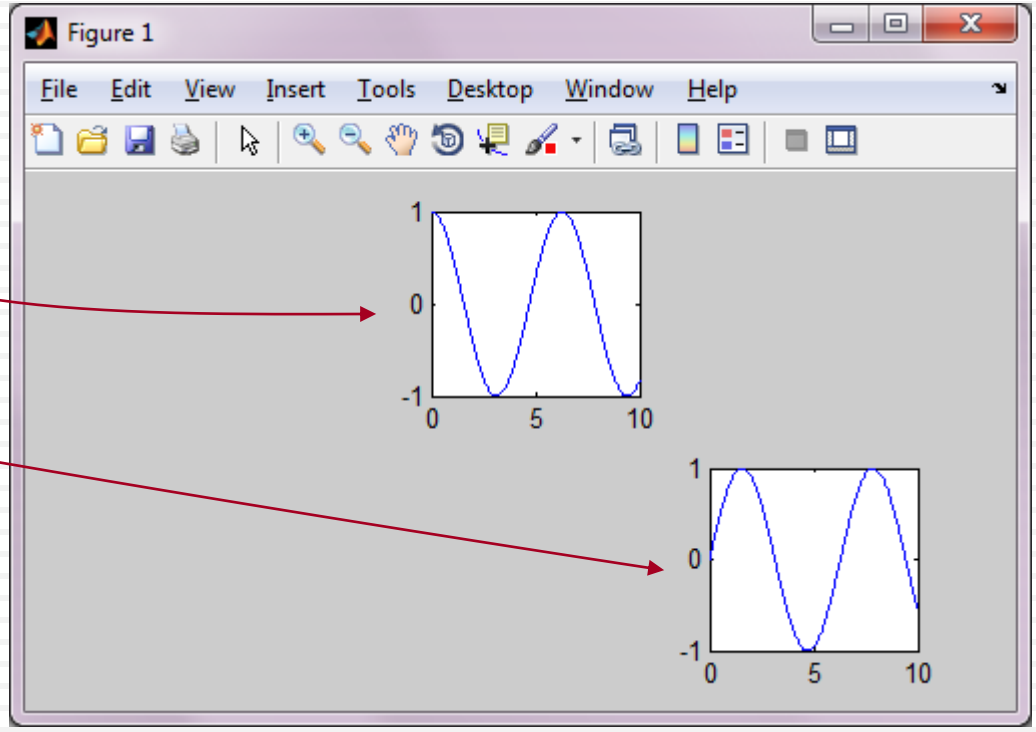
Prikaz više dijagrama u istom prozoru

- subplot(m,n,k): izbor k-tog dijagrama u matrici dijagrama sa m-redova i n-kolona



Prikaz više dijagrama u istom prozoru

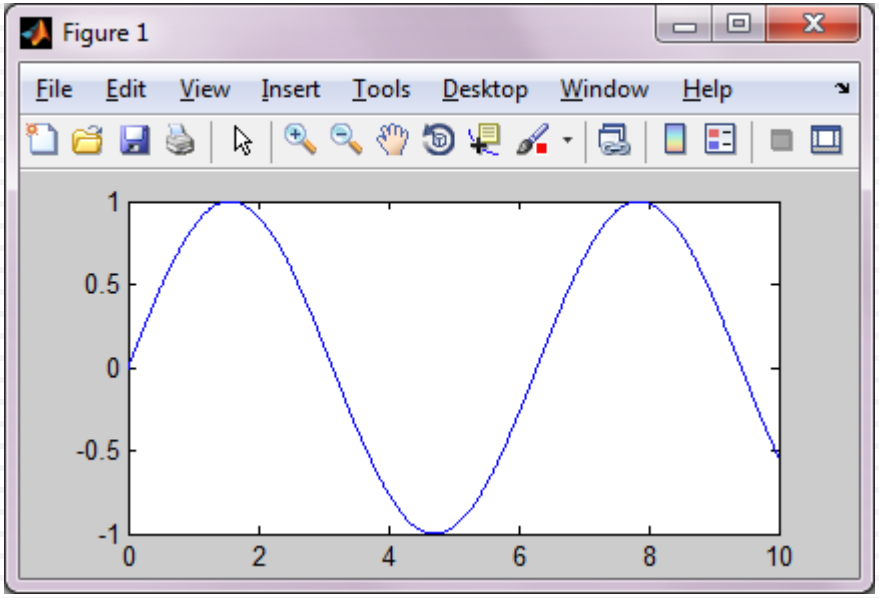
```
>> x=linspace(0,10,1000);
>> subplot(2,3,6);
>> plot(x,sin(x));
>> subplot(2,3,2);
>> plot(x,cos(x)); ...
```

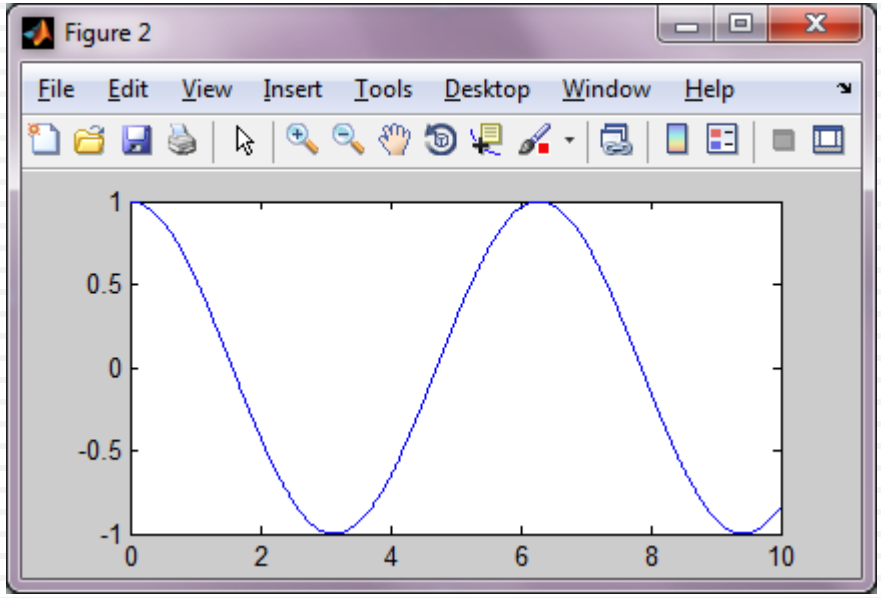


Prikaz više dijagrama u različitim prozorima

- figure(k): izbor/otvaranje k-tog prozora.
- close(k): zatvara prozor k

```
>> x=linspace(0,10,1000);
>> figure(1);
>> plot(x,sin(x));
>> figure(2);
>> plot(x,cos(x));
```

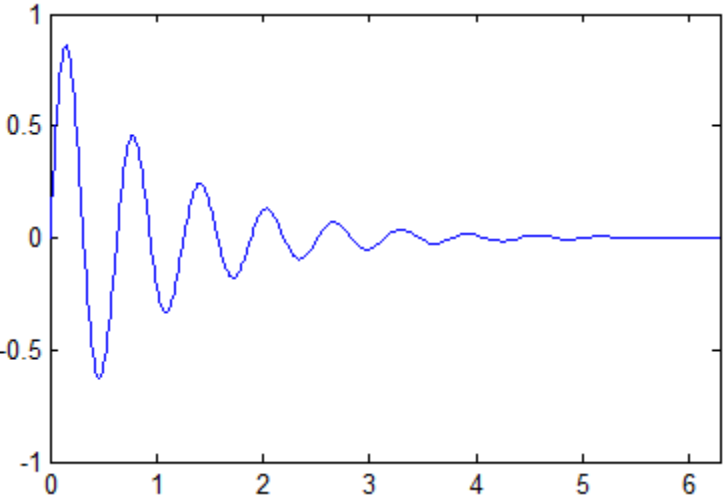




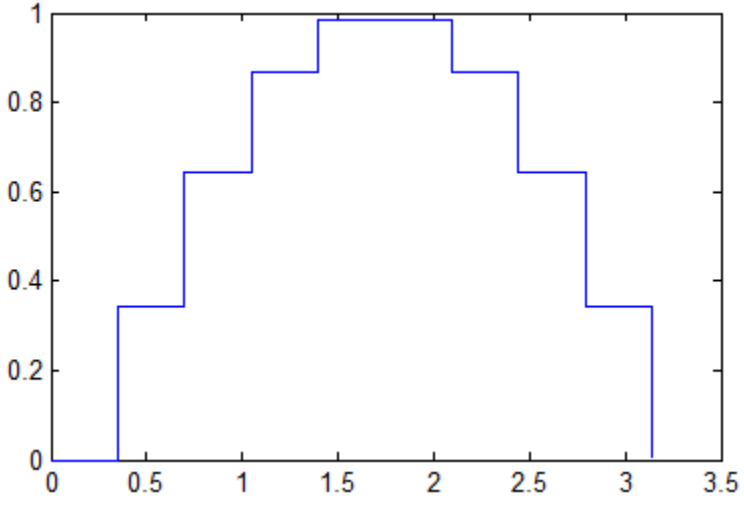
Posebni 2D dijagrami

- fplot: crta funkciju zadanu analitičkim izrazom (automatski korak)
- stairs: stepenasti grafik
- polar: prikaz u polarnim koordinatama
- bar: prugasti dijagram
- hist: skup vrednosti vektora se deli u grupe, a zatim se prikazuje broj elemenata u okviru grupe
- pie: relativan odnos pojedinog elementa vektora prema celini

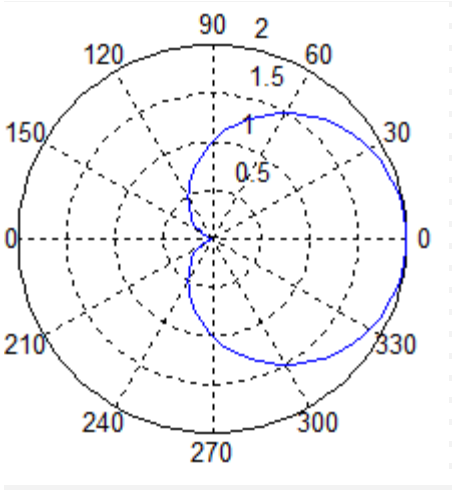
```
fplot('exp(-x)*sin(10*x)',[0,2*pi]);
```



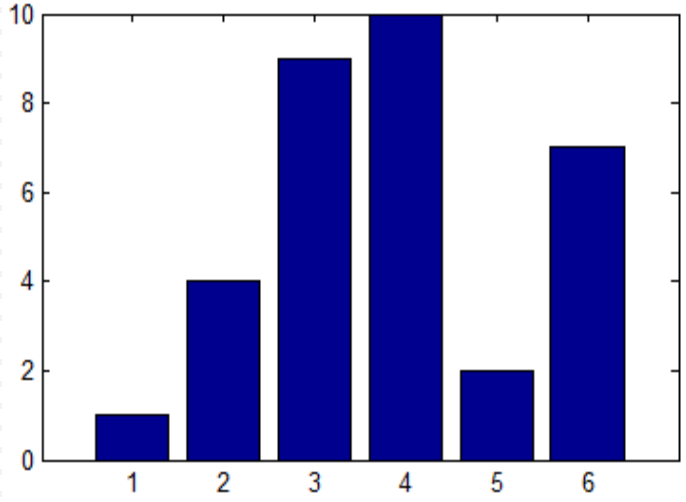

```
x=linspace(0,pi,10);
stairs(x,sin(x));
```



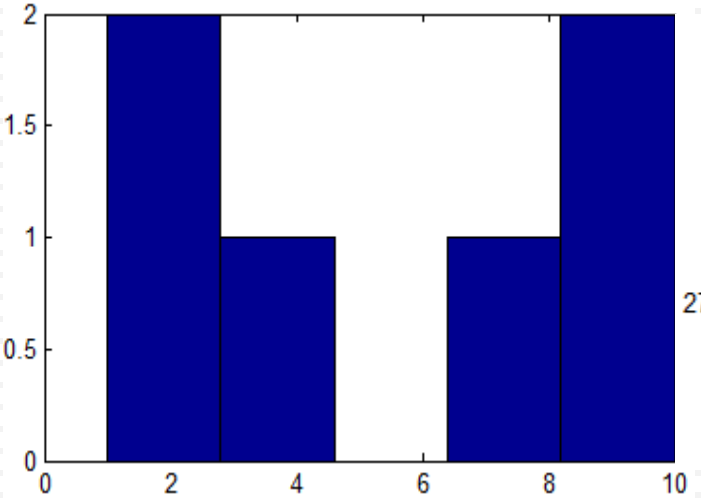

```
phi=0:.1:2*pi;
ro=1+cos(phi);
polar(phi,ro);
```



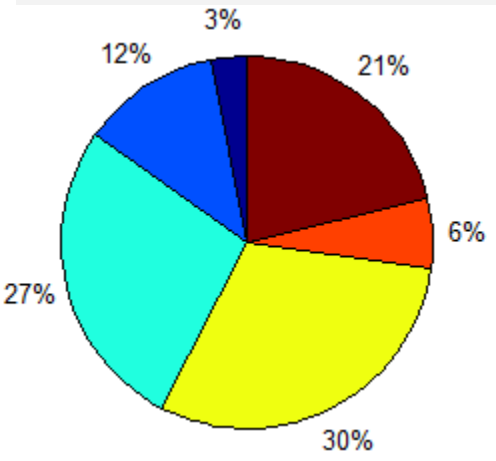

```
x=[1 4 9 10 2 7];
bar(x);
```




```
x=[1 4 9 10 2 7];
hist(x,5);
```




```
x=[1 4 9 10 2 7];
pie(x);
```



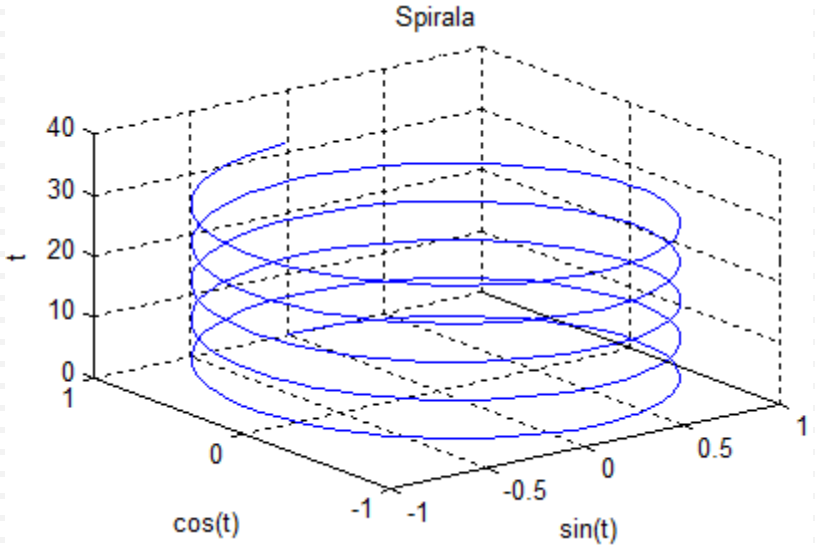
3D Grafika

- Crtanje parametarski zadate linije u prostoru
  - plot3(x,y,z)
- Crtanje površi z=f(x,y)
  - mesh(x,y,z): mrežasti model
  - surf(x,y,z): solid model

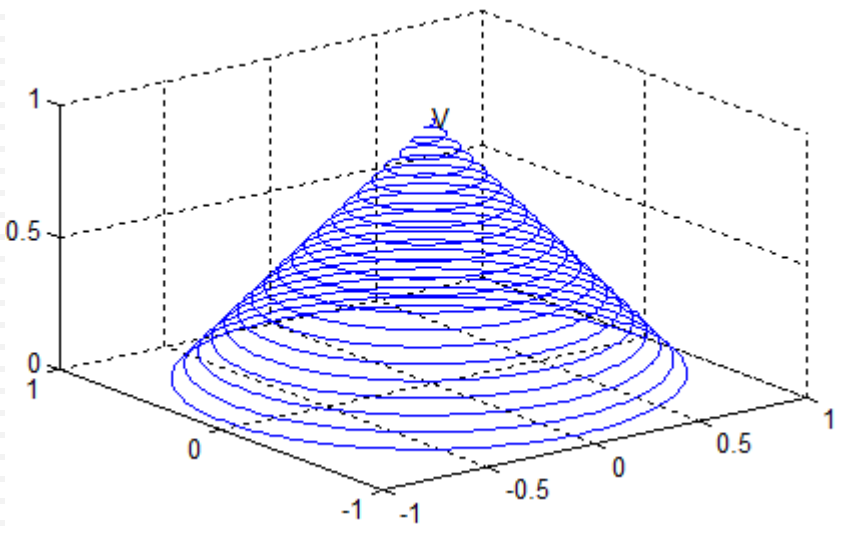
Crtanje linije u prostoru

- plot3: Uopštenje funkcije plot2

```
>> t=0:0.1:10*pi;
>> plot3(sin(t),cos(t),t);
>> title('Spirala');
>> xlabel('sin(t)');
>> ylabel('cos(t)');
>> zlabel('t');
>> grid on;
```

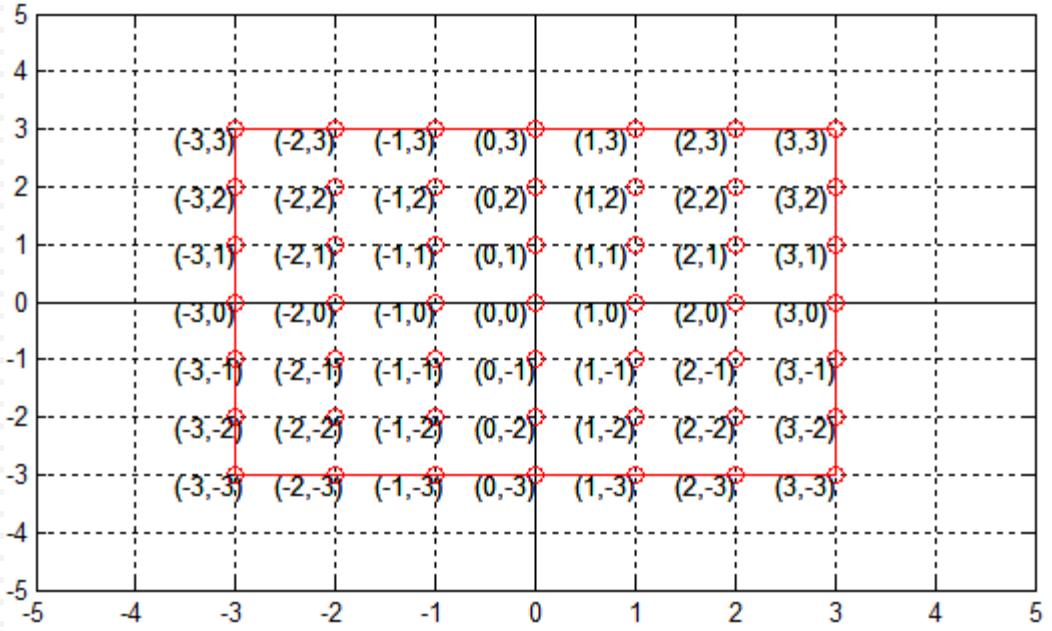



```
>> turns=40*pi;
>> theta=linspace(0,turns,4000);
>> x=cos(theta).*(turns-theta)./turns;
>> y=sin(theta).*(turns-theta)./turns;
>> z=theta./turns;
>> plot3(x,y,z);
>> grid on
>> text(0,0,1,'V');
```



Crtanje površi

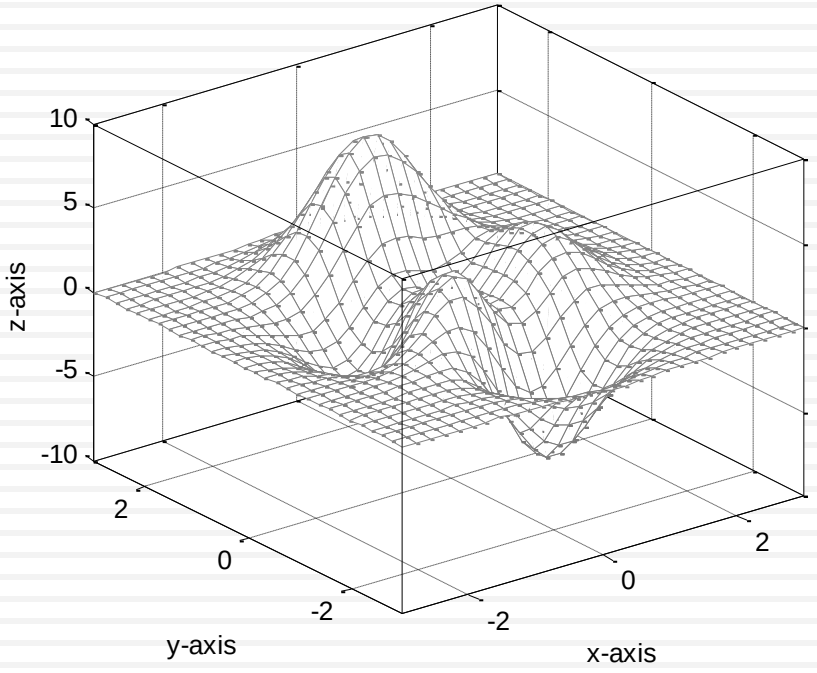
- z=f(x,y)
  - Formira se podela (grid) tačaka u ravni (x,y)



  - U tačkama podele (x,y) izračunavaju se vrednosti funkcije
  - Tačke podele i vrednosti funkcije formiraju mrežu tačaka u prostoru (x,y,f(x,y)) kroz koju se crta površ

Formiranje podele

- Potrebno je formirati podelu (grid) tačaka u ravni u kojima će se izračunavati vrednosti funkcije z=f(x,y)




```
x =-3  -2  -1  0  1  2  3
y =-3  -2  -1  0  1  2  3
z=f(x,y)
```

Nije dobro: Potrebne su sve vrednosti od y za svaku vrednost x i obratno, kako bi izračunali funkciju definisanu na oblasti

Ovo je u redu: odgovarajući elementi od xx i yy formiraju podelu sa tačkama u kojima treba izračunavati vrednosti funkcije

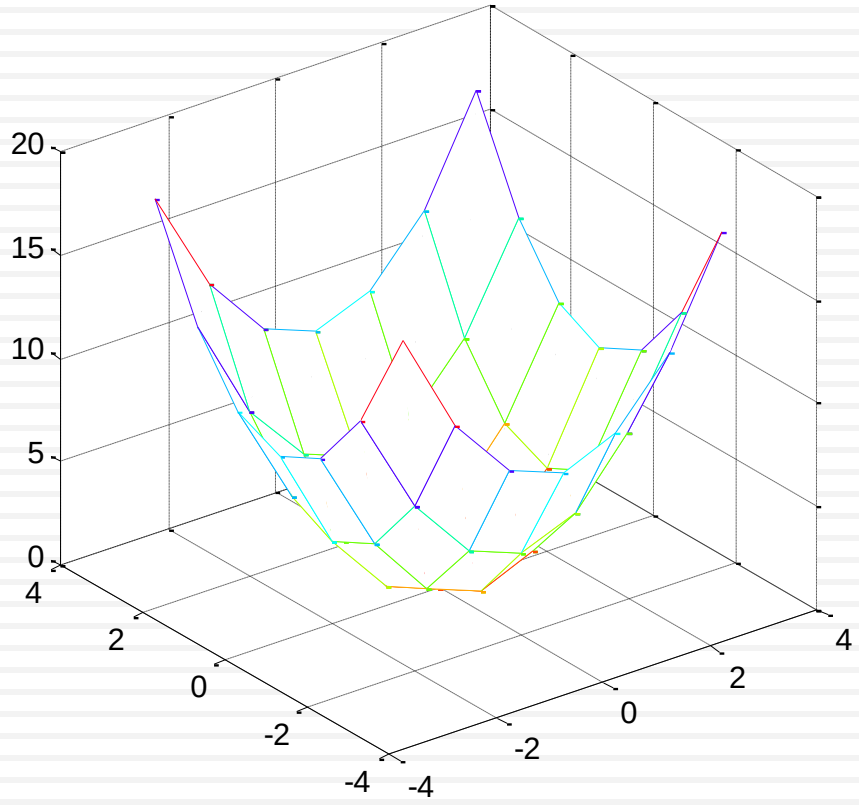
```
xx =-3  -2  -1  0  1  2  3
    -3  -2  -1  0  1  2  3
    -3  -2  -1  0  1  2  3
    -3  -2  -1  0  1  2  3
    -3  -2  -1  0  1  2  3
    -3  -2  -1  0  1  2  3
    -3  -2  -1  0  1  2  3

yy =-3  -3  -3  -3  -3  -3  -3
    -2  -2  -2  -2  -2  -2  -2
    -1  -1  -1  -1  -1  -1  -1
     0   0   0   0   0   0   0
     1   1   1   1   1   1   1
     2   2   2   2   2   2   2
     3   3   3   3   3   3   3
z=f(xx,yy)
```

Vrednosti xx se menjaju duž VRSTA
Vrednosti yy se menjaju duž KOLONA

meshgrid, mesh Funkcije

- $z = (x^2 + y^2)$ u oblasti $-3 \leq x \leq 3,\ -3 \leq y \leq 3$

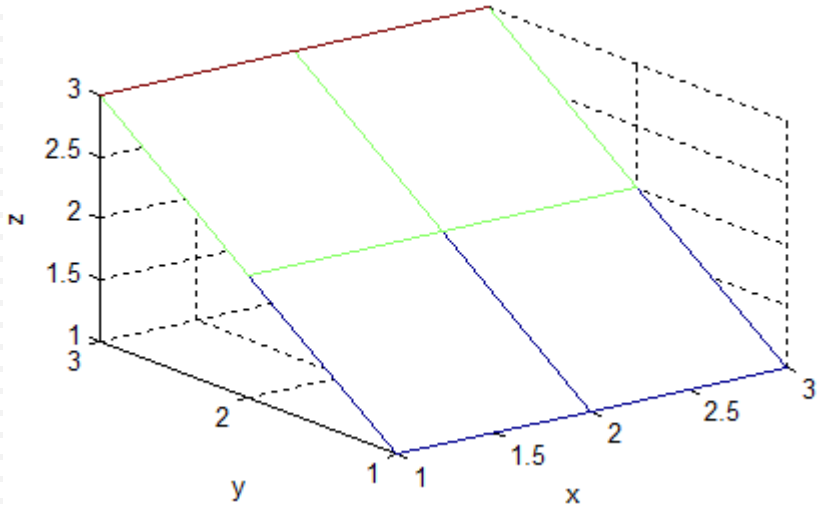
```
>> x=-3:3;
>> y=-3:3;
>> [xx,yy]=meshgrid(x,y)
xx =
     3    -2    -1     0     1     2     3
    -3    -2    -1     0     1     2     3
    -3    -2    -1     0     1     2     3
    -3    -2    -1     0     1     2     3
    -3    -2    -1     0     1     2     3
    -3    -2    -1     0     1     2     3
    -3    -2    -1     0     1     2     3
yy =
    -3    -3    -3    -3    -3    -3    -3
    -2    -2    -2    -2    -2    -2    -2
    -1    -1    -1    -1    -1    -1    -1
     0     0     0     0     0     0     0
     1     1     1     1     1     1     1
     2     2     2     2     2     2     2
     3     3     3     3     3     3     3
>> zz=xx.^2 + yy.^2;
>> mesh(xx,yy,zz)
```



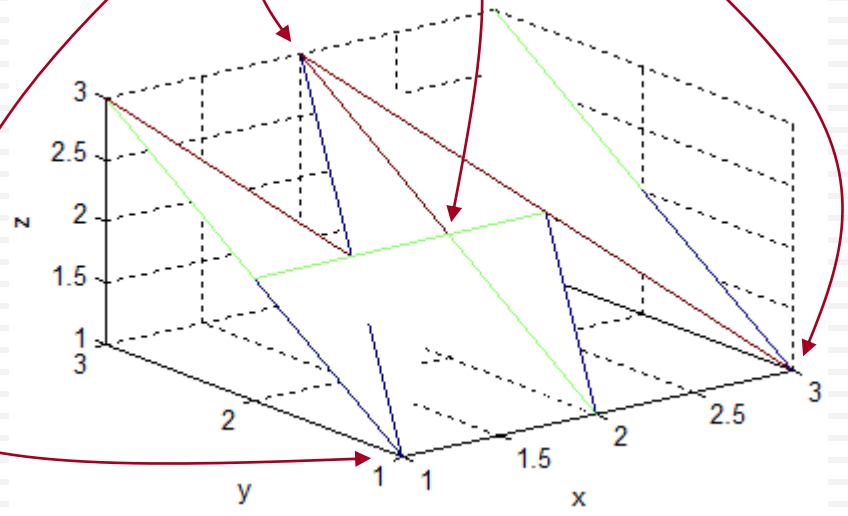
meshgrid Funkcija

- Loše formirana podela rezultuje pogrešnim spajanjem tačaka mreže

```
xx =
     1   2   3
     1   2   3
     1   2   3

yy =  1   1   1
      2   2   2
      3   3   3
>> zz=yy;
>> mesh(xx,yy,zz)
```




```
xx =
     1   2   3
     1   2   3
     1   2   3

yy =  1   3   1
      2   2   2
      3   1   3
>> zz=yy;
>> mesh(xx,yy,zz)
```



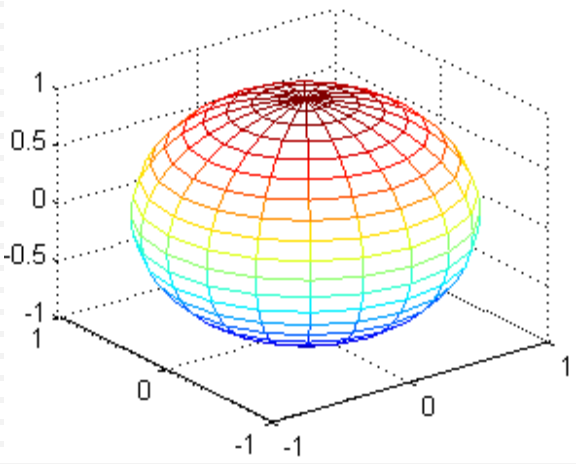
sphere, ellipsoid, cylinder Funkcije

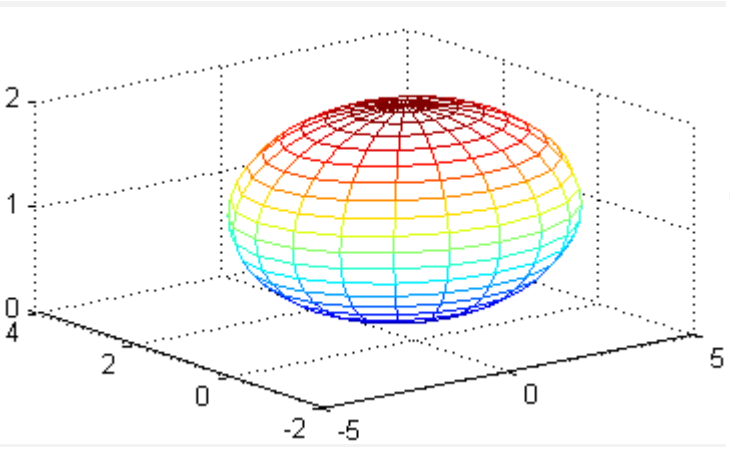
- Formiraju mreže tačaka odgovarajućih površi (podrazumevan broj tačaka je 20)
  - [x,y,z]=sphere(n): Formira mrežu sa n tačaka jedinične sfere
  - [x,y,z]=ellipsoid(xc,yc,zc,a,b,c,n): Formira mrežu sa n tačaka elipsoida sa centrom u (xc,yc,zc) i poluosama a,b,c
  - [x,y,z]=cylinder(R,n): Formira mrežu cilindra sa n tačaka u osnovi, visine 1 i centrom u koordinatnom početku. Osnove cilindra su kružnice čiji su poluprečnici zadani vektorom R

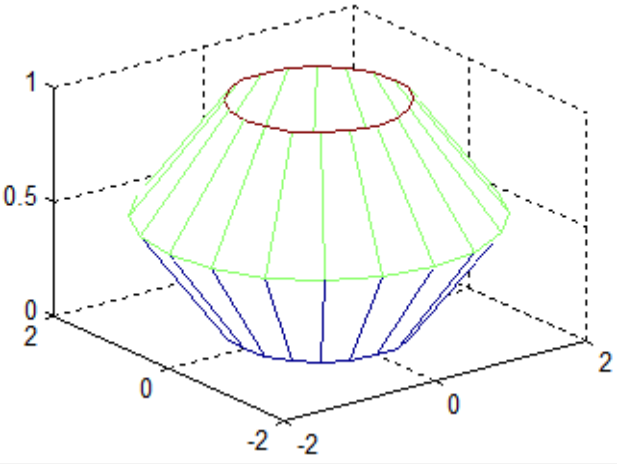
```
>> [X,Y,Z]=sphere;
>> mesh(X,Y,Z);
```

```
>> [X,Y,Z]=ellipsoid(1,1,1,4,2,1);
>> mesh(X,Y,Z);
```

```
>> [X,Y,Z]=cylinder(1,2,1);
>> mesh(X,Y,Z);
```





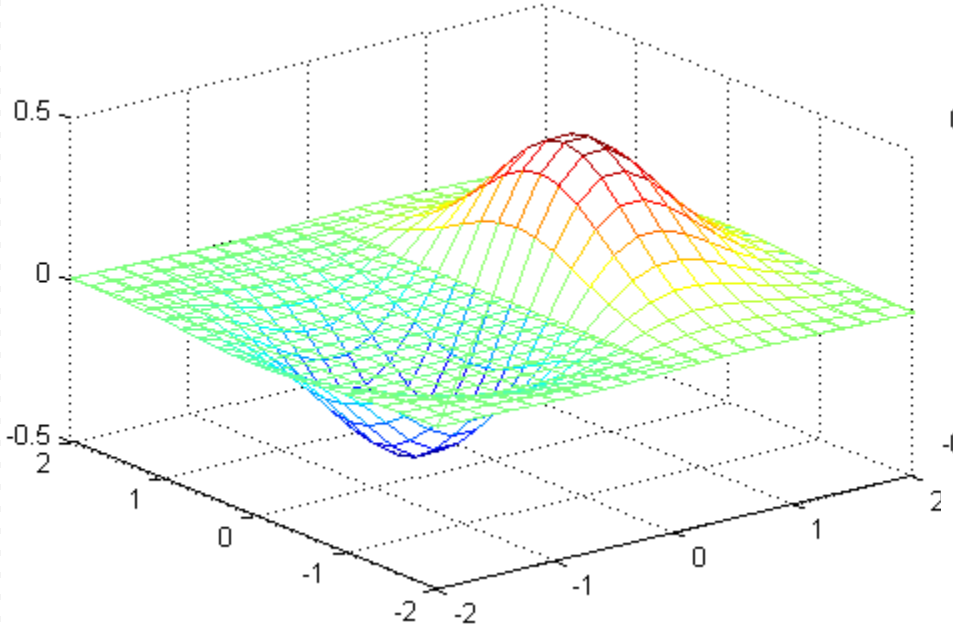


surf Funkcija

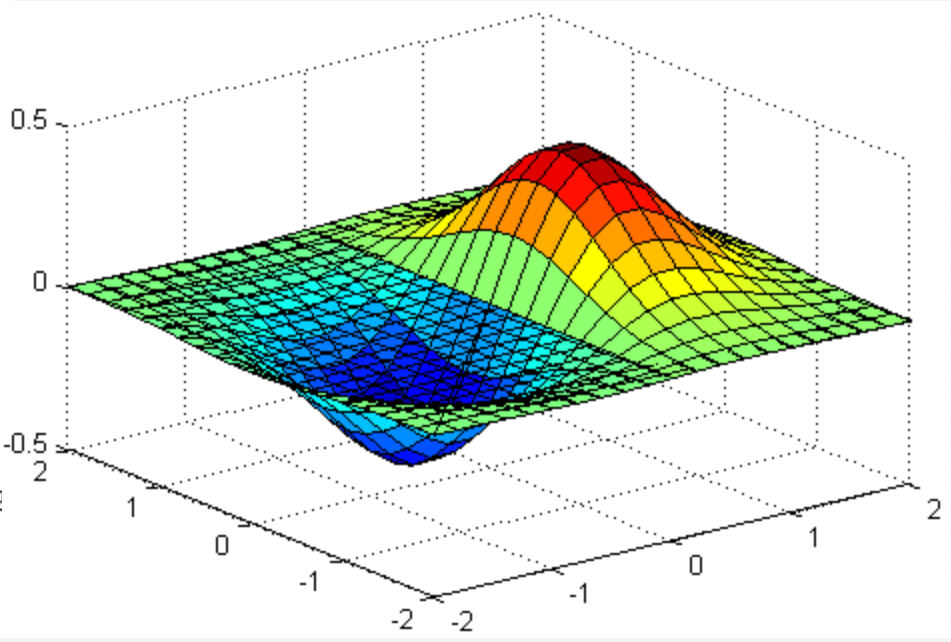
- Popunjava mrežasti grafik površinama (facets) sa 3 ili 4 temena (solid model). Podrazumevano senčenje shading faceted

```
>> x=-2:.2:2;
>> y=-2:.2:2;
>> [X,Y] = meshgrid(x,y);
>> Z=X.*exp(-X.^2 - Y.^2);
```

```
>> mesh(X,Y,Z);
```




```
>> surf(X,Y,Z);
```

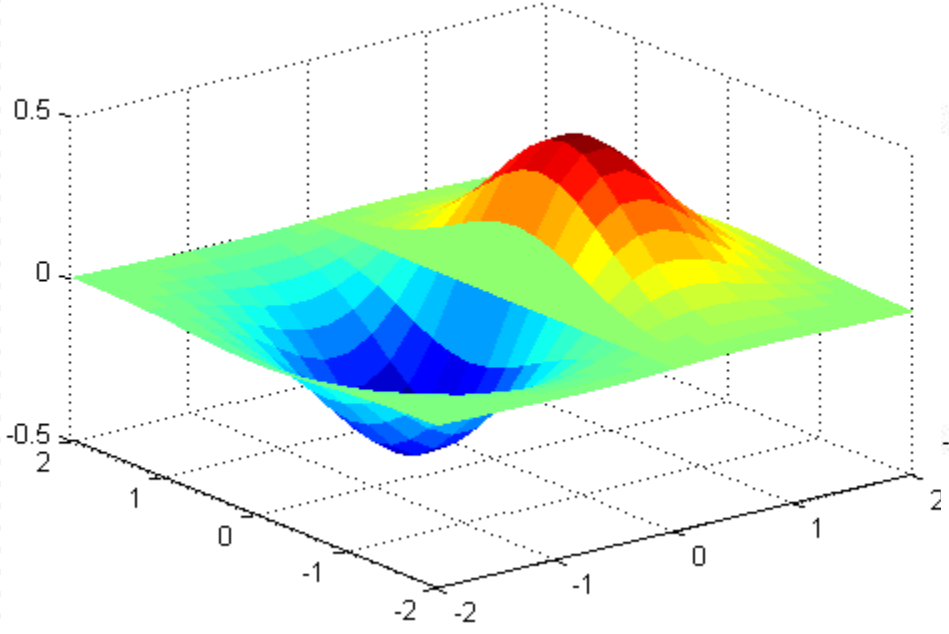


Senčenje

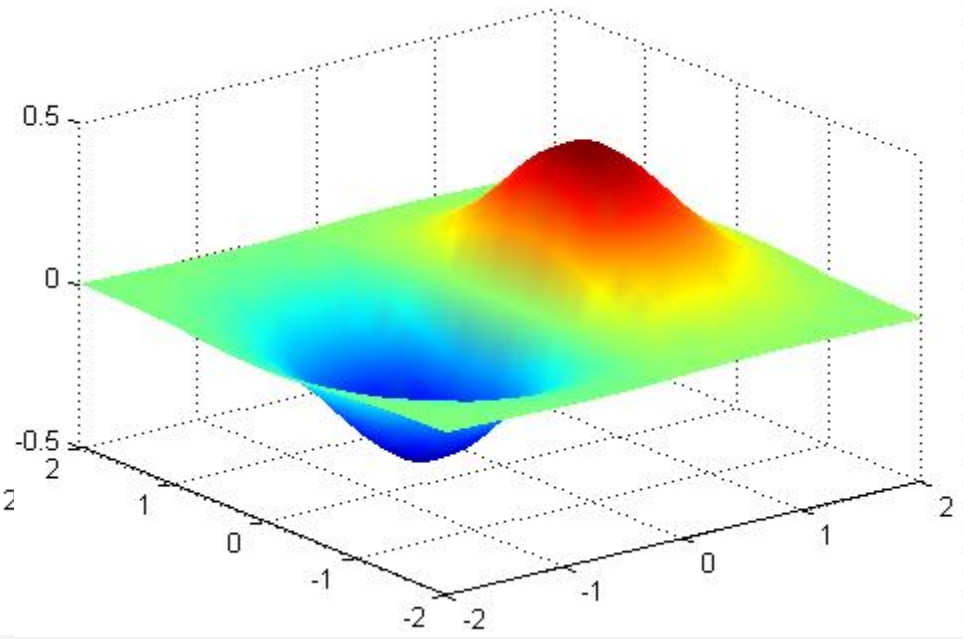
- shading flat: ukida mrežu i svaki od facet-a je u jednoj boji
- shading interp: interpolira boju duž facet-a

```
>> x=-2:.2:2;
>> y=-2:.2:2;
>> [X,Y] = meshgrid(x,y);
>> Z=X.*exp(-X.^2 - Y.^2);
>> surf(X,Y,Z);
```

```
>> shading flat;
```



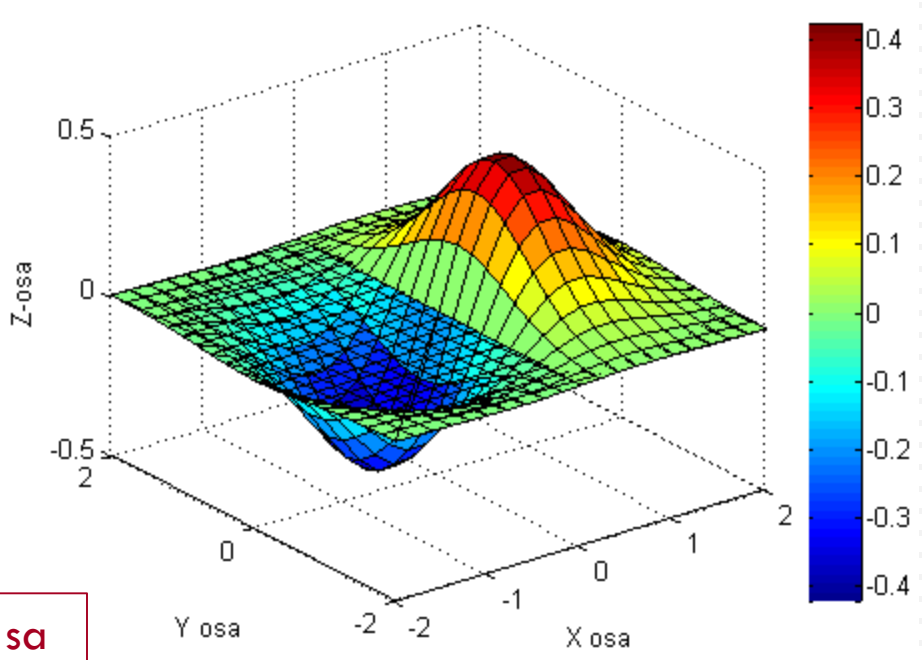

```
>> shading interp;
```



colorbar Funkcija

- colorbar: Prikazuje skalu boja koje se koriste za prikaz slike

```
>> x=-2:.2:2;
>> y=-2:.2:2;
>> [X,Y] = meshgrid(x,y);
>> Z=X.*exp(-X.^2 - Y.^2);
>> surf(X,Y,Z);
>> xlabel('X osa')
>> ylabel('Y osa')
>> zlabel('Z-osa')
>> colorbar;
```



Boja se podrazumevano menja sa promenom vrednosti z-koordinate

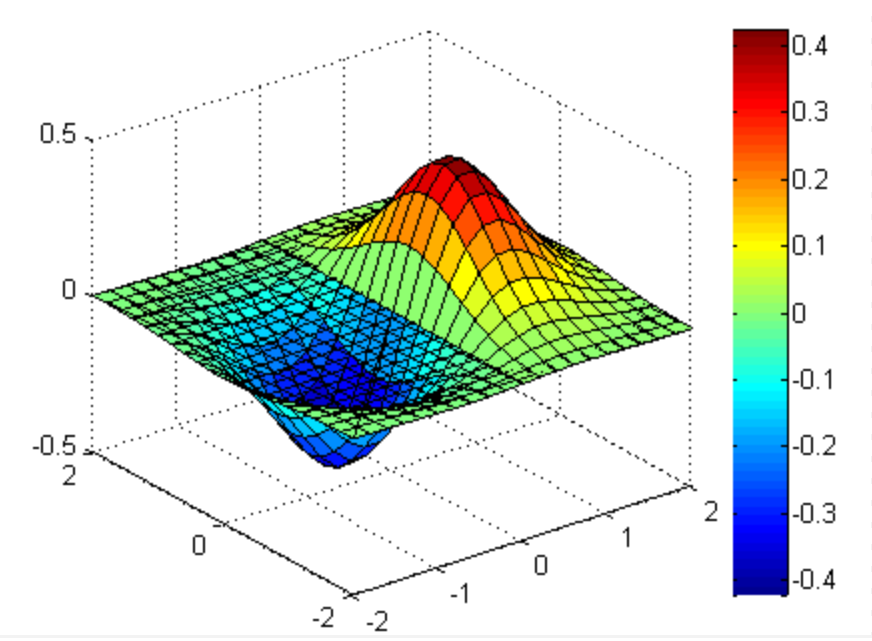
colormap Funkcija

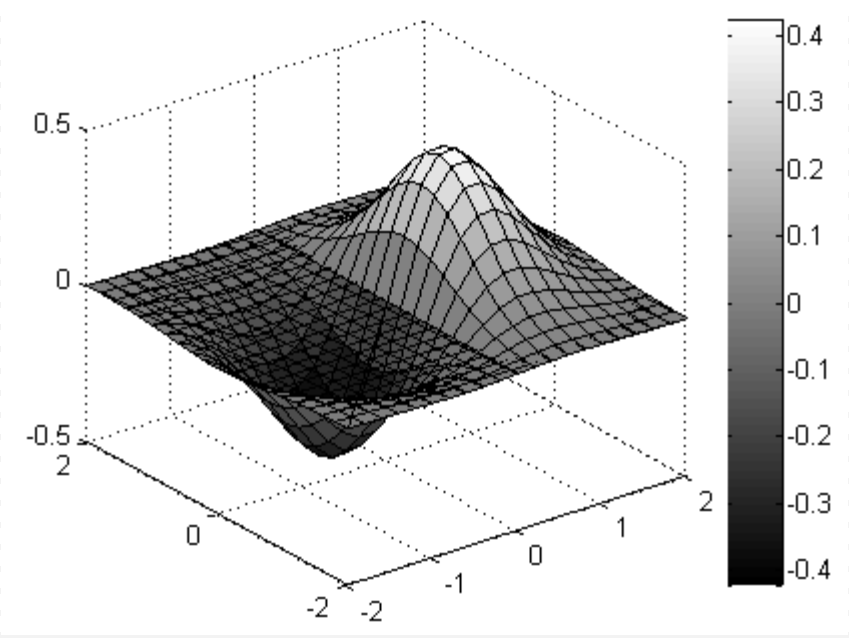
- colormap: Svakom grafičkom prozoru se pridružuje mapa boja
  - Matrica formata mx3
  - Svaki red reprezentuje jednu boju koja se reprezentuje količinama osnovnih boja (crvene, zelene i plave) i čije su vrednosti u prvoj drugoj odnosno trećoj koloni
    - [0,0,0] – crna
    - [1,0,0] – crvena
    - [1,1,1] – bela
  - Neke od ugrađenih mapa boja su: jet, hsv, gray, hot, cool, …
  - colormap(C): pridružuje tekućem grafičkom prozoru mapu boja C
  - colormap default: Tekućem grafičkom prozoru pridružuje podrazumevanu mapu boja (jet)

colormap Funkcija

```
>> x=-2:.2:2;
>> y=-2:.2:2;
>> [X,Y] = meshgrid(x,y);
>> Z=X.*exp(-X.^2 - Y.^2);
>> surf(X,Y,Z);
```

```
>> colormap(gray);
```



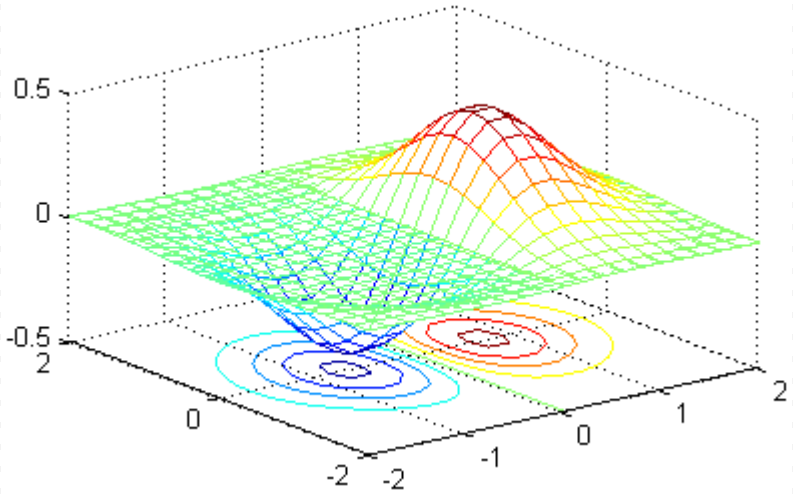


Konturni grafici

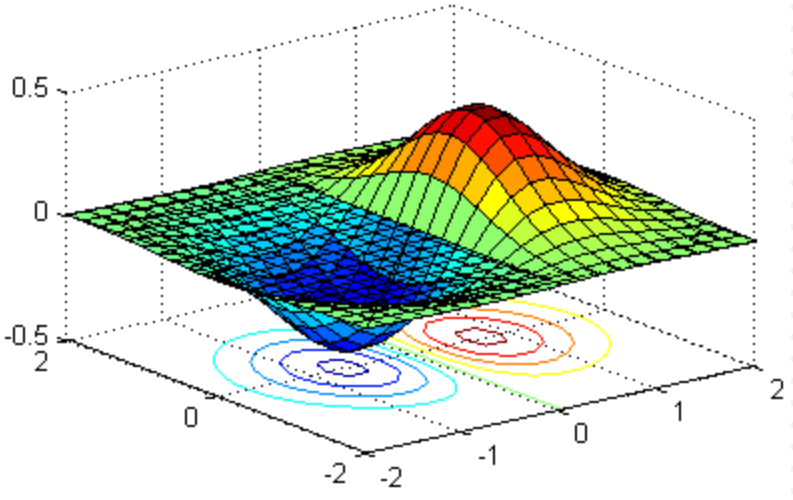
- meshc, surfc: Crta se i konturni grafik ispod površi

```
>> x=-2:.2:2;
>> y=-2:.2:2;
>> [X,Y] = meshgrid(x,y);
>> Z=X.*exp(-X.^2 - Y.^2);
```

```
>> meshc(X,Y,Z);
```




```
>> surfc(X,Y,Z);
```

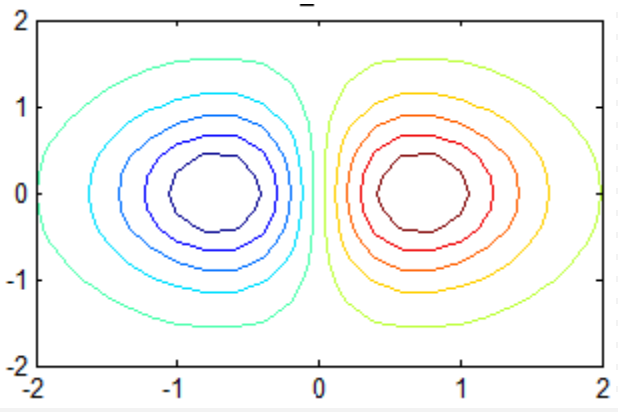


Konturni grafici

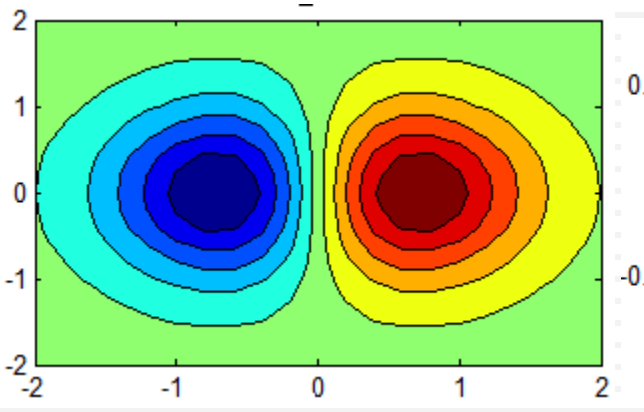
- Crta n nivoskih linija površi (ako se n ne zada određuje se automatski).
  - contour(x,y,z,n): Crta prost konturni grafik (n nivoskih linija)
  - contourf(x,y,z,n): Crta popunjeni konturni grafik
  - contour3(x,y,z,n): Crta konturni grafik u 3D

```
>> [X,Y] = meshgrid(-2:.2:2,-2:.2:2);
>> Z=X.*exp(-X.^2 - Y.^2);
```

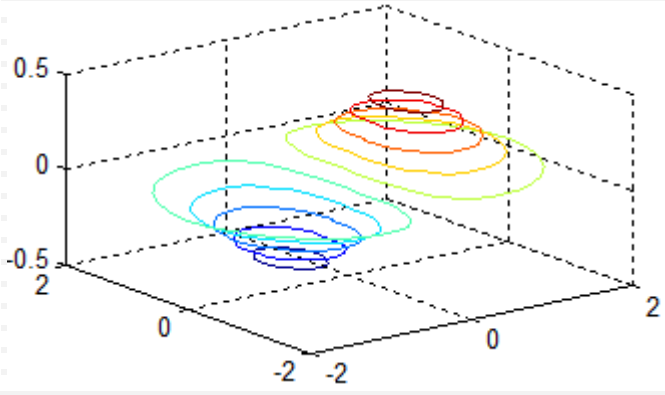
```
>> contour(X,Y,Z,10);
```




```
>> contourf(X,Y,Z,10);
```

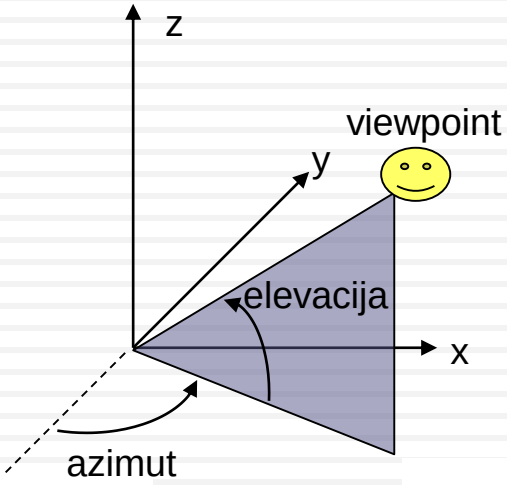


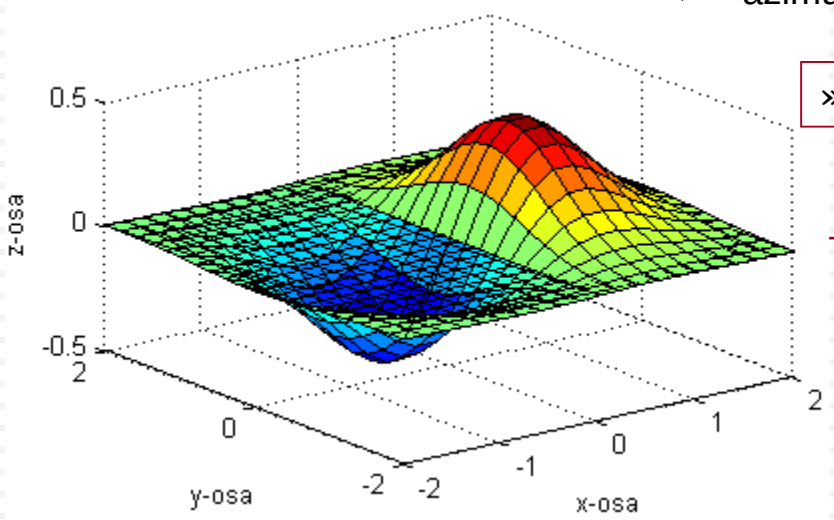

```
>> contour3(X,Y,Z,10);
```



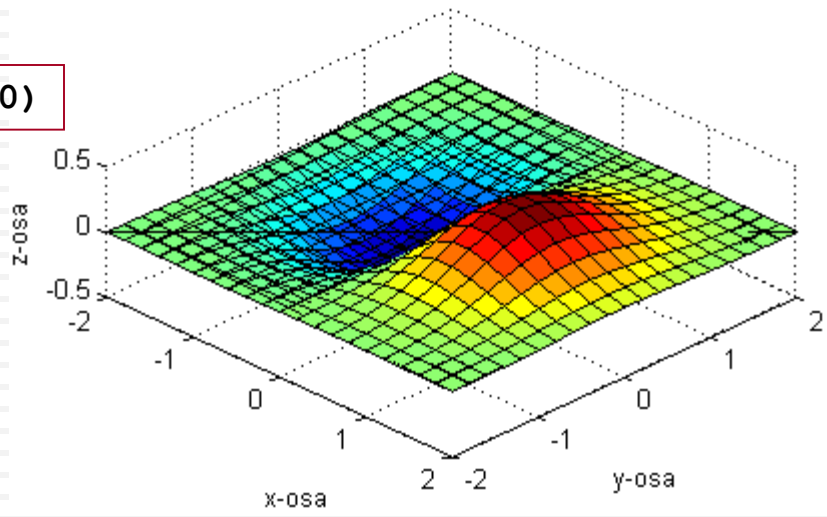
Promena ugla gledanja

- view([x,y,z]): Pravac gledanja je dat vektorom [x,y,z]
- view(az,el)



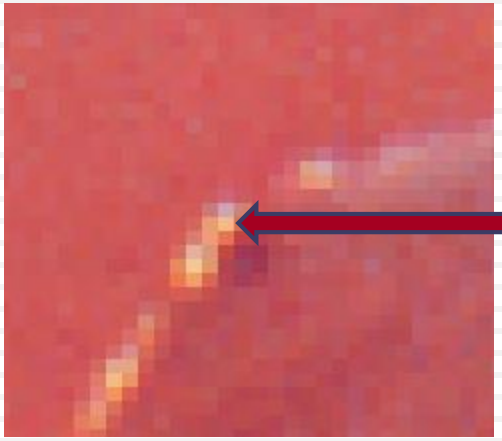



```
» view(45,60)
```



Obrada slike

- U Matlab-u se slika reprezentuje matricom
- Elementi matrice su pikseli



- Sve operacije nad matricama se mogu koristiti i nad slikama

Matrica slike

y-osa

x-osa

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & \\ \dots & & & \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

a_{ij} je intenzitet na skali-sive, piksela u i-toj vrsti i j-toj koloni

Tipovi slika u Matlab-u

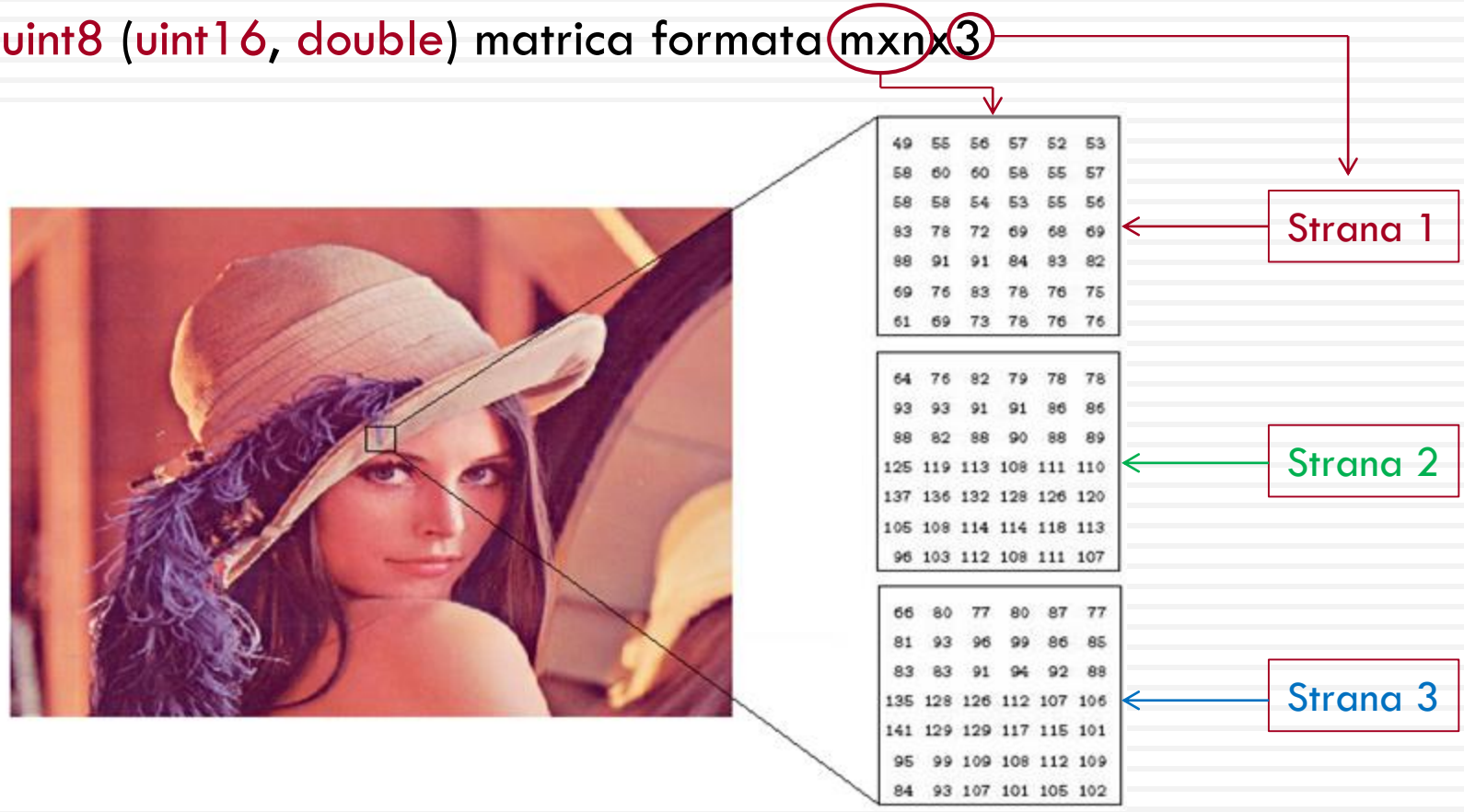
- Binarna (Crno-Bela) slika – Svaki piksel je crn ili beo, odnosno reprezentuje se sa {0,1}
  - logical matrica formata mxn

Tipovi slika u Matlab-u

- Slika sa skalom-sive – Svaki piksel je intenzitet sive, obično se reprezentuje vrednostima od 0 (crna) do 255 (bela)
  - uint8 (uint16) matrica formata mxn
  - Binarna slika je specijalan slučaj slike sa skalom-sive koja ima dve nijanse

Tipovi slika u Matlab-u

- RGB slika – Svaki piksel je neke boje koja se reprezentuje količinom svake od osnovnih boja (R-crvene, G-zelene i B-plave). Ove količine su vrednosti od 0 do 255
  - uint8 (uint16, double) matrica formata mxnx3



Tipovi slika u Matlab-u

- Indeksirane slike – Uglavnom na slici u koloru nisu zastupljene sve boje, pa se može uštedeti memorijski prostor ako se koristi mapa boja. Za svaki piksel beleži se indeks u mapi boja
  - uint8 (uint16, double) matrica formata mxn

Konverzija tipa slike

- im2bw – konverzija u binarnu sliku
- gray2ind – konverzija slike sa skalom-sive u indeksiranu sliku
- ind2gray – konverzija indeksirane slike u sliku sa skalom-sive
- rgb2gray – konverzija RGB slike u sliku sa skalom-sive
- rgb2ind – konverzija RGB slike u indeksiranu sliku
- im2double – konvertuje matricu slike u double matricu
- im2uint16 – konvertuje matricu slike u uint16 matricu
- im2uint8 – konvertuje matricu slike u uint8 matricu
- …

Učitavanje slike

- Mogu se učitavati slike različitih formata
  - BMP (Microsoft Windows Bitmap)
  - GIF (Graphics Interchange Files)
  - HDF (Hierarchical Data Format)
  - JPEG (Joint Photographic Experts Group)
  - PCX (Paintbrush)
  - PNG (Portable Network Graphics)
  - TIFF (Tagged Image File Format)
  - …

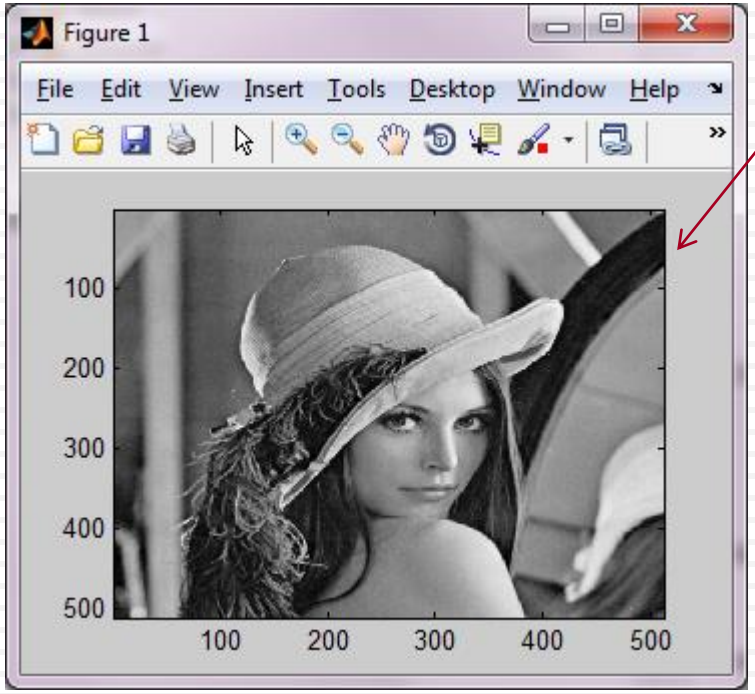
imread Funkcija

- imread('ime datoteke')
  - Učitava slike sa skalom-sive, RGB i indeksirane slike iz datoteke sa datim imenom
  - Slika se učitava u uint8 matricu odgovarajućeg formata
  - Automatski određuje format slike na osnovu informacija iz zaglavlja datoteke
    - Opciono može se navesti i format kao drugi argument

imshow Funkcija

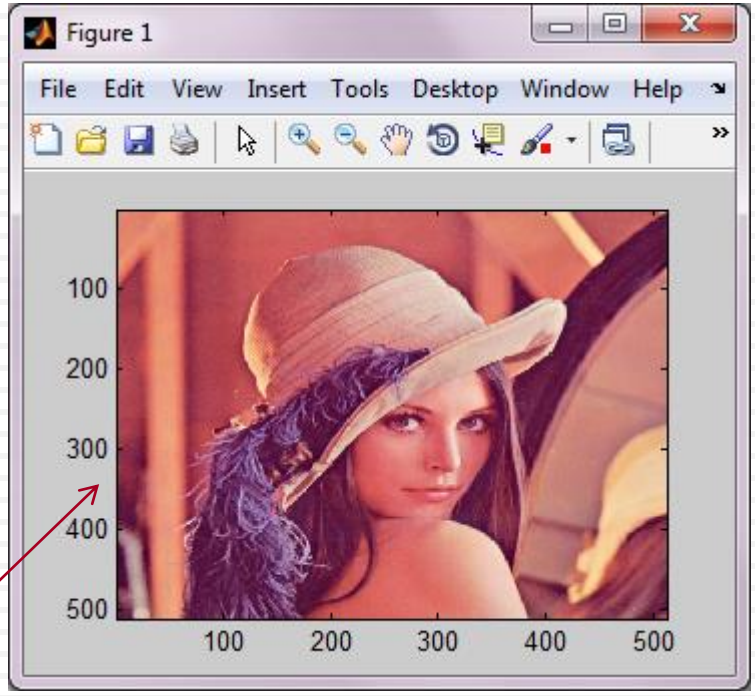
```
>> imat=imread('lenags.bmp');
>> whos imat
  Name        Size              Bytes  Class    Attributes
  imat        512x512           262144 uint8
>> imshow(imat);
```

```
>> imat=imread('lena.bmp');
>> whos imat
  Name        Size              Bytes  Class    Attributes
  imat        512x512x3         786432 uint8
>> imshow(imat);
```



Skala-Sive

RGB



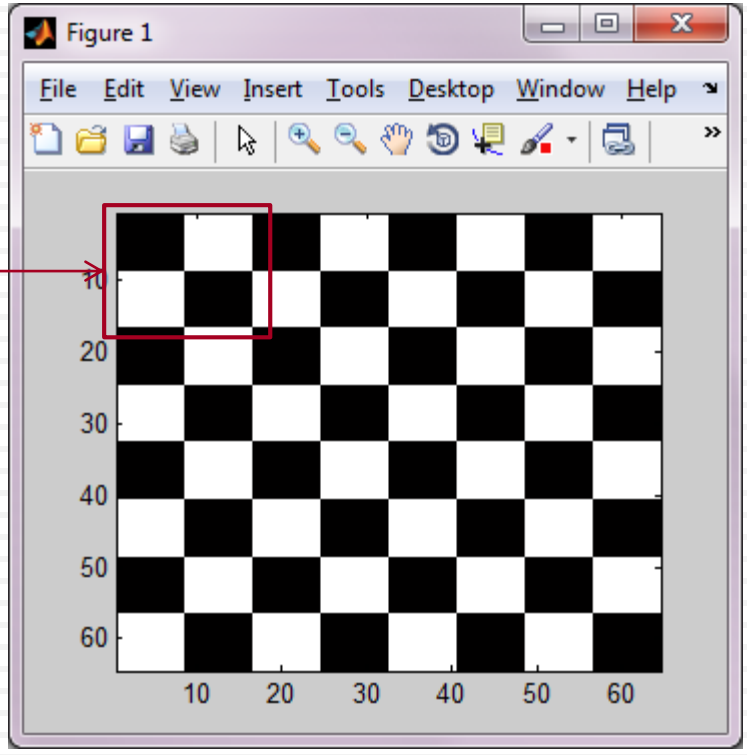
Upis slike u datoteku

- Mogu se upisivati slike u različitim formatima
- imwrite(imat,'ime datoteke')
  - Upisuje slike sa skalom-sive, RGB i indeksirane slike reprezentovane uint8 matricom *imat* u datoteku sa datim imenom zajedno sa odgovarajućim zaglavljem i kolor mapom ako je potrebno
  - Automatski određuje format slike na osnovu ekstenzije datoteke u koju se vrši upis
    - Opciono može se navesti i format kao treći argument

Primer

- Crno bela slika šahovske table formata 256x256 piksela

```
>> c=logical(zeros(32));
>> b=logical(ones(32));
>> cb_bc=[c b;b c];
>> sah=repmat(cb_bc, 4, 4);
>> imshow(sah);
>> imwrite(sah, 'sah.bmp');
```

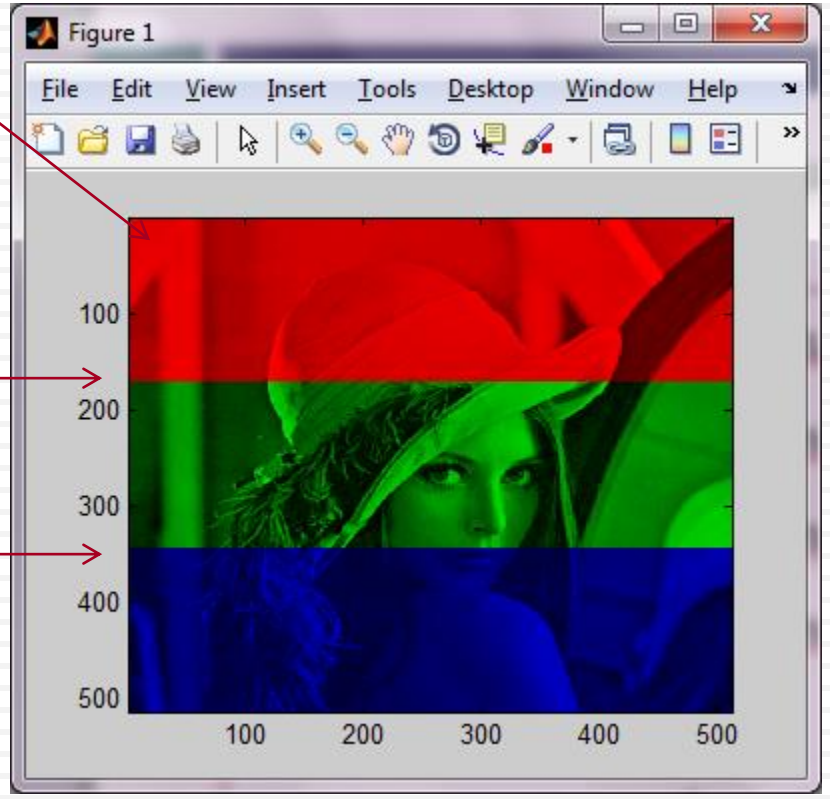


Primer

- Gornja trecina slike ima samo crvenu komponentu, srednja samo zelenu a poslednja trećina samo plavu komponentu

```
>> imat=imread('lena.bmp');
>> rows=size(imat,1);
>> trecina1=fix(rows/3);
>> imat(1:trecina1,:, 2:3)=0;
>> trecina2=fix(2*rows/3);
>> imat(trecina1+1:trecina2,:, [1,3])=0;
>> imat(trecina2+1:end,:, 1:2)=0;
>> imshow(imat);
>> imwrite(imat, 'lena.bmp');
```

Količina zelene i plave je 0



Primer

- Simetrija u odnosu na vertikalnu osu

```
>> imat=imread('lena.bmp');
>> imat(:,:,1)=fliplr(imat(:,:,1));
>> imat(:,:,2)=fliplr(imat(:,:,2));
>> imat(:,:,3)=fliplr(imat(:,:,3));
>> imshow(imat);
>> imwrite(imat, 'anel.bmp');
```