



Naučni softverski alati

MATLAB/Octave – Elementarne
operacije sa matricama i poljima
brojeva; 2D grafika

Matrične operacije

- sabiranje **+** - matrice moraju biti istih dimenzija
- oduzimanje **-** - matrice moraju biti istih dimenzija
- množenje ***** - matrice moraju imati jednake unutrašnje dimenzije

■ Primjer:

```
>> A=2*ones(3,5);B=2*eye(3,5);C=A+B,D=A-B
```

C =

4	2	2	2	2
2	4	2	2	2
2	2	4	2	2

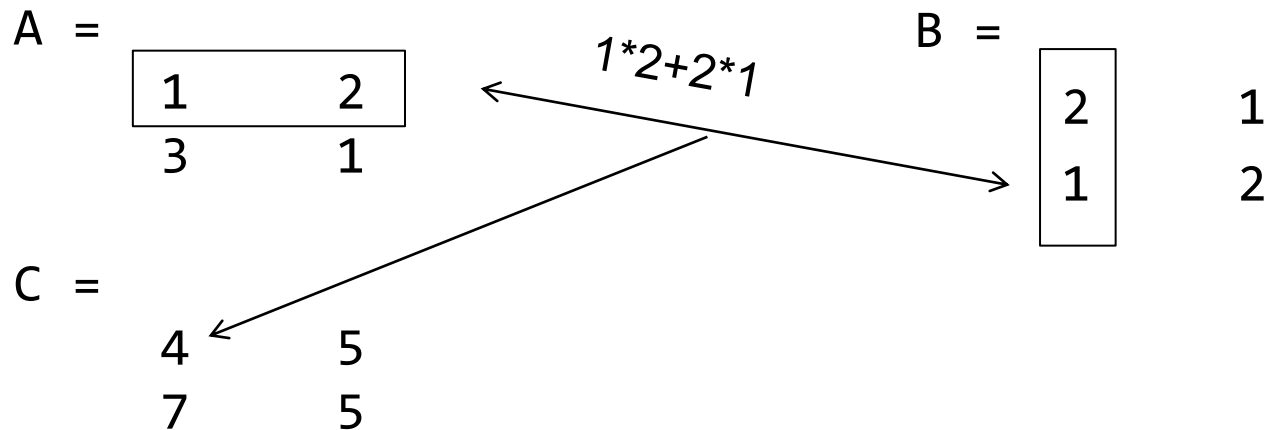
D =

0	2	2	2	2
2	0	2	2	2
2	2	0	2	2

Matrično množenje ...

■ Primjer:

>> $A = [1 \ 2; 3 \ 1], B = [2 \ 1; 1 \ 2], C = A * B$



>> $x=[1, 2, 3]; y=[4, 7, 8]; x1=x*y', x2=x'*y$

x1 =

42

x2 =

4	7	8
8	14	16
12	21	24

Matrične operacije – primjeri

Sabiranje, oduzimanje i množenje matrica

```
>> a = [1 2; 3 4], b = [2 2; 2 2]
```

```
a =
```

```
 1  2  
 3  4
```

```
b =
```

```
 2  2  
 2  2
```

```
>> c = a + b, d = a - b, e = a * b
```

```
c =
```

```
 3  4  
 5  6
```

```
d =
```

```
-1  0  
 1  2
```

```
e =
```

```
 6  6  
14 14
```

matrično množenje dvije matrice



Operacije nad poljem brojeva

- Operacije nad elementima matrice (tačka ispred operatora!!!):
 - množenje **.*** - matrice moraju imati iste dimenzije

- Primjer:

```
>> A=[1 2; 3 1], B=[2 1; 1 2], C = A .* B
```

A =

1	2
3	1

B =

2	1
1	2

C =

2	2
3	2

```
>> x = [1, 2, 3]; y = [4, 7, 8]; x1 = x .* y
```

x1 =

4	14	24
---	----	----

Matrično dijeljenje i dijeljenje polja brojeva

- dijeljenje s lijeva $/$ ($X=B/A$ je rješenje sistema $X * A = B$ – broj kolona mora biti isti)
 - dijeljenje s desna \backslash ($X=A\backslash B$ je rješenje sistema $A * X = B$ – broj vrsta mora biti isti)
- Operacije nad elementima matrice (tačka ispred operatora!!!):
 - dijeljenje s lijeva $./$
 - dijeljenje s desna $.\backslash$

■ Primjer:

>> $A = [1 \ 2; 3 \ 1]$, $B = [2 \ 1; 1 \ 2]$, $C = A ./ B$, $D = A .\ B$

A =		C =	
	1		0.5000
	2		2.0000
	3		3.0000
	1		0.5000
B =		D =	
	2		2.0000
	1		0.5000
	1		0.3333
	2		2.0000

Stepenovanje

- stepenovanje $^{\wedge}$ - matrica mora biti kvadratna
- Operacije nad elementima matrice (tačka ispred operatora!!!):
 - stepenovanje $.^{\wedge}$ - matrica ne mora biti kvadratna
- Primjer

```
>> a = [1 2; 3 4],
```

```
>> h = a .^ 2, g = a / 2, f = 2 ./ a
```

```
a =
```

```
    1    2  
    3    4
```

```
h =
```

```
    1    4  
    9   16
```

```
g =
```

```
    0.5000    1.0000  
    1.5000    2.0000
```

```
f =
```

```
    2.0000    1.0000  
    0.6667    0.5000
```

Kada je skalar u imeniocu, nije potrebna tačka ispred operatora

Matrice i funkcije

- Elementarne matrice funkcije:
 - determinanta matrice **det(x)**
 - inverzina matrica **inv(x)**
- Funkcija se poziva njenim imenom, tj. imenom m-fajla.
- Ako funkcija ima parametre navode se u malim zagradama () razdvojeni zarezima.
- Ako funkcija vraća više vrijednosti, promjenljive koje ih prihvataju se navode u uglastim zagradama [] razdvojene zarezima. Redoslijed parametara je bitan, jer se dodjela rezultata vrši s lijeva na desno.
- Primjer:

```
a = [1, 2; 3, 4; 5, 6];
```

```
[m, n] = size(a);
```

m = 3 broj vrsta, n = 2 broj kolona

Rješavanje sistema jednačina u MATLAB-u

- Neka je zadat sistem jednačina

$$x + 2y - z = 2$$

$$2x + z = 5$$

$$x - y + z = 8$$

$$AX = Y$$

$$A = \begin{bmatrix} 1 & 2 & -1 \\ 2 & 0 & 1 \\ 1 & -1 & 1 \end{bmatrix} \quad X = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad Y = \begin{bmatrix} 2 \\ 5 \\ 8 \end{bmatrix}$$

- Ovaj sistem se može predstaviti matrično, i riješiti korišćenjem matričnog računa kao $X = A^{-1}Y$.

```
>> A = [1 2 -1; 2 0 1; 1 -1 1];
```

```
>> Y = [2; 5; 8];
```

```
>> X = inv(A) * Y
```

```
X =
```

```
13
```

```
-16
```

```
-21
```

Elementarne matematičke funkcije

1. Argument se tretira kao polje brojeva. Ako je argument matrica:

$$X = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{m1} & X_{m2} & \cdots & X_{mn} \end{bmatrix}$$

rezultat funkcije $f(X)$ će biti matrica:

$$f(X) = \begin{bmatrix} f(X_{11}) & f(X_{12}) & \cdots & f(X_{1n}) \\ f(X_{21}) & f(X_{22}) & \cdots & f(X_{2n}) \\ \vdots & \vdots & \ddots & \vdots \\ f(X_{m1}) & f(X_{m2}) & \cdots & f(X_{mn}) \end{bmatrix}$$

2. X može imati realne ili kompleksne elemente.
3. Složeni izrazi i funkcije se grade pomoću elementarnih funkcija i operacija za manipulaciju poljima brojeva $+$, $-$, \cdot , $/$, \backslash , \wedge

Elementarne matematičke funkcije

- korjenovanje: **sqrt(x)** \sqrt{x}
- apsolutna vrijednost: **abs(x)** $|x|$
- fazni stav: **angle(x)**
- realni dio: **real(x)**
- imaginarni dio: **imag(x)**
- konjugovanje: **conj(x)**
- prirodni logaritam: **log(x)** $\ln x$
- logaritam sa osnovom 10: **log10(x)** $\log_{10} x$
- eksponencijalna funkcija: **exp(x)** e^x
- trigonometrijske funkcije (**očekuju argument zadat u radijanima!!!**)
sin(x), cos(x), tan(x), cot(x), asin(x), acos(x), atan(x),
acot(x)

Elementarne matematičke funkcije

- najbliži cio broj: **round(x)**
- veći cio broj: **ceil(x)**
- manji cio broj: **floor(x)**
- cijeli dio broja: **fix(x)**

```
>> A = [1.6 2.4; -1.6 -2.4];
```

```
A =
```

```
    1.6000    2.4000  
   -1.6000   -2.4000
```

```
>>A1 = round(A), A2 = ceil(A), A3 = floor(A), A4 = fix(A)
```

```
A1 =
```

```
    2    2  
   -2   -2
```

```
A2 =
```

```
    2    3  
   -1   -2
```

```
A3 =
```

```
    1    2  
   -2   -3
```

```
A4 =
```

```
    1    2  
   -1   -2
```

Elementarne matematičke funkcije

- najveći zajednički djelilac: **gcd(a, b)**
- najmanji zajednički sadržalac: **lcm(a, b)**
- ostatak pri dijeljenju: **rem(a, b)**
- obrtanje redosljeda vrsta: **flipud(A)**
- obrtanje redosljeda kolona: **flip1r(A)**

Primjer: Neka nam je zadatak da napravimo tablicu vrijednosti sinusa i kosinusa za sve uglove od 0 do 180 stepeni sa korakom od 15 stepeni. Jedan od načina rješavanja ovog zadatka je:

```
>> x = (0 : 15 : 180)';  
>> y = x * pi / 180; %obavezno pretvoriti stepene u radijane  
>> T = [x, sin(y), cos(y)]
```

Crtanje 2D grafika funkcija

- **plot(y)** – crtanje vektora y u zavisnosti od rednog broja elementa.

Grafik se dobija tako što se pravim linijama povežu susjedne tačke (n, y_n) i $(n+1, y_{n+1})$, $n = 1, 2, \dots, N-1$

- **plot(x, y)** – crtanje funkcije y u zavisnosti od nezavisno promjenljive x .

Grafik se dobija tako što se pravim linijama povežu susjedne tačke (x_n, y_n) i (x_{n+1}, y_{n+1}) , $n = 1, 2, \dots, N-1$. Podrazumijeva se Dekartov pravougaoni koordinatni sistem.

- **plot(x1, y1, x2, y2)** – crtanje više funkcija na istom grafiku

- Prvim izvršenjem funkcija `plot` otvara novi grafički prozor i crta grafik. Svako naredno izvršenje postojeći grafik mijenja sa novim.

- Primjer: Nacrtati grafik funkcije $\sin x$, za $-\pi \leq x \leq \pi$

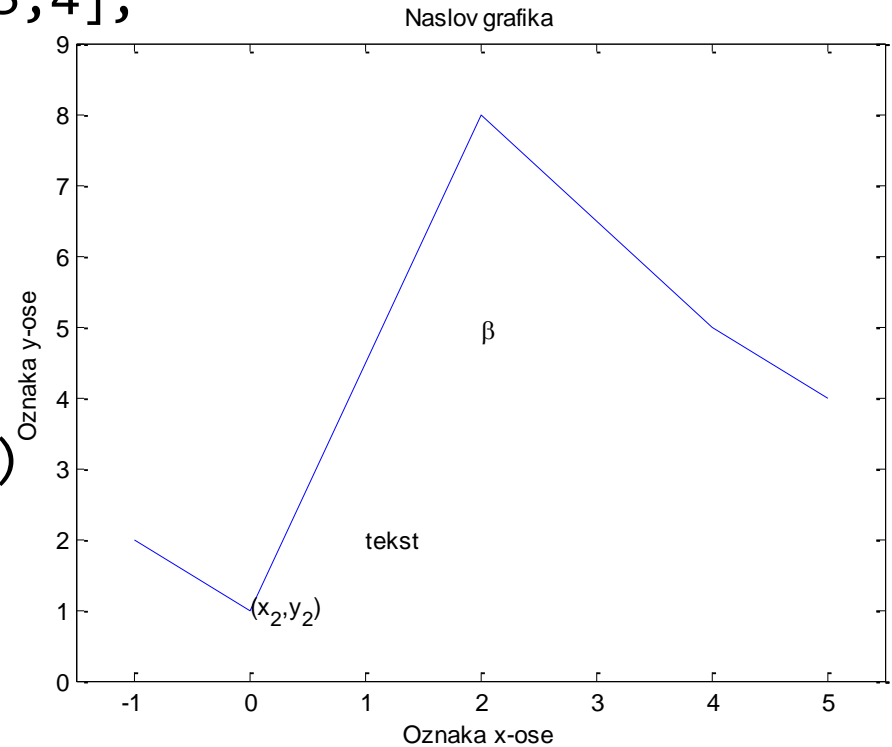
```
>> x = -pi:pi/20:pi;  
>> y = sin(x);  
>> plot(x,y)
```

Označavanje osa i grafika

- **xlabel('tekst')** – postavlja tekst kao oznaku za x-osu
- **ylabel('tekst')** – postavlja tekst kao oznaku za y-osu
- **title('tekst')** – postavlja tekst kao naslov grafika
- **text(x, y, 'tekst')** – postavlja tekst na poziciju zadatu preko x i y.
- Na grafiku se mogu nalaziti i oznake koje u sebi imaju indeks. Indeks se navodi nakon povlake. Tako bi bilo: **'x_1'** → x_1 , **'x_2'** → x_2 ,
'x_{i,j}^{12}' → x_{ij}^{12}
- Ukoliko želimo kao oznake imati grčka slova, koristimo `\` prije naziva željenog grčkog slova. Tako će biti: **'\alpha'** → α , **'\beta'** → β
- Komandom **axis([Xmin, Xmax, Ymin, Ymax])** definišemo granice vidljivog dijela grafika. Istom komandom možemo postići dodatne efekte, na primjer jednaku skalu na x i y osi, uklanjanje osa sa grafika, jednake ose, vraćanje osa ... **axis equal**, **axis off**, **axis square**, **axis on**, **axis auto**, **axis normal**

Primjer

```
>> x=[-1 0 2 4 5];y=[2,1.0,8,5,4];  
>> plot(x,y)  
>> title('Naslov grafika')  
>> xlabel('Oznaka x-ose')  
>> ylabel('Oznaka y-ose')  
>> text(1,2,'tekst')  
>> text(x(2),y(2),'(x_2,y_2)')  
>> text(2,5,'\beta')  
>> axis([-1.5,5.5,0,9])
```



Vrste linija i oznaka

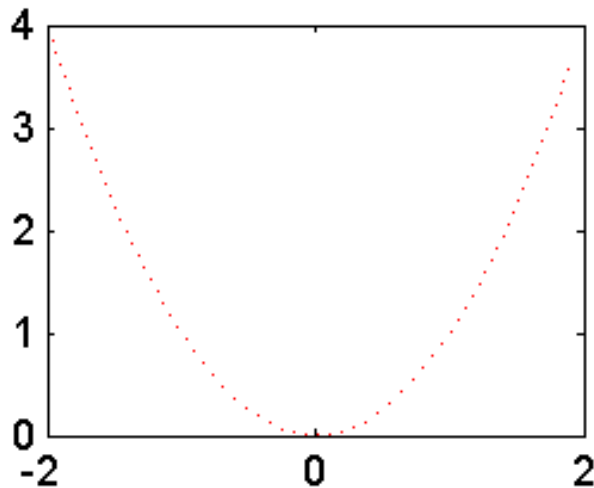
Tip linije		Tip simbola	
-	puna linija	o	kružić
- .	crtta tačka	x	krstić
:	tačkasta	*	zvjezdica
--	isprekidana	+	plus

Boje	
y	Žuta (yellow)
r	Crvena (red)
g	Zelena (green)
b	Plava (blue)
k	Crna (black)
w	Bijela (white)
m	Magenta (magenta)
c	Cijan (cyan)

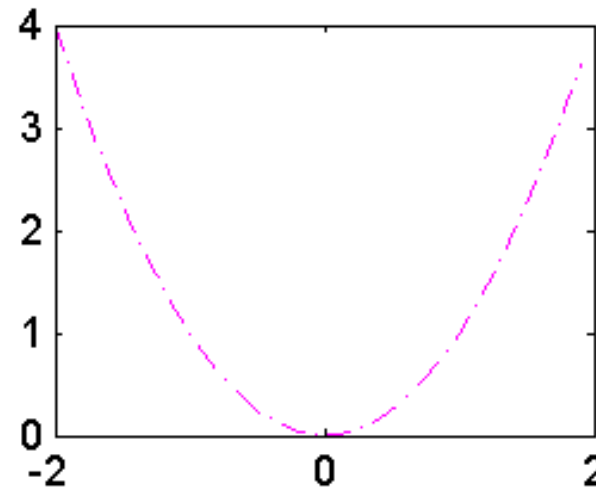
```
>> x = -pi : pi / 20 : pi;  
>> y = sin(x);  
>> plot(x, y, 'r:')  
crtanje crvenom isprekidanom  
linijom
```

Primjeri

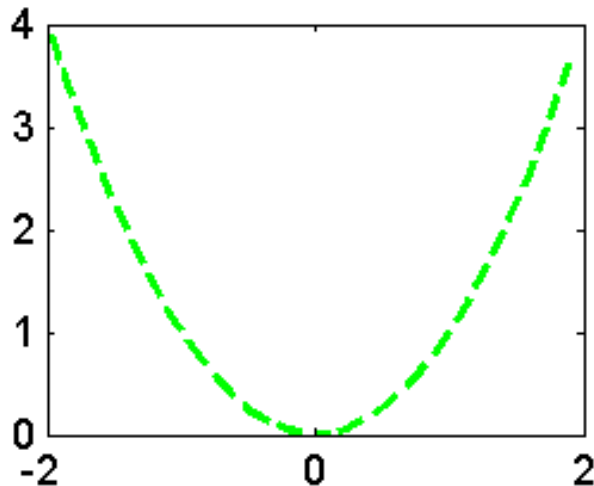
`plot(x,y,'r:')`



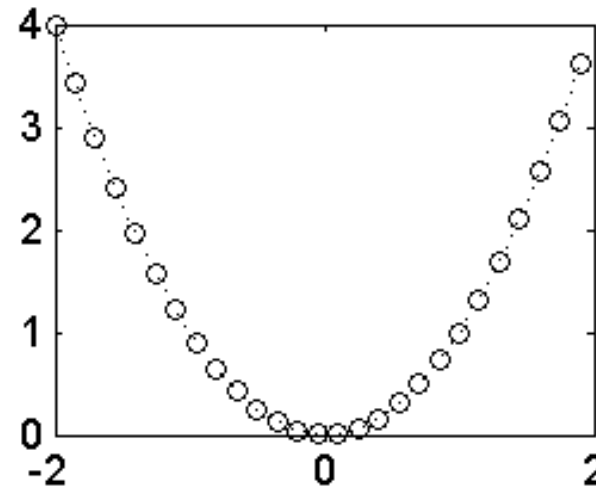
`plot(x,y,'m-.'`



`plot(x,y,'g--', 'LineWidth', 2)`



`plot(x,y,'k:o')`



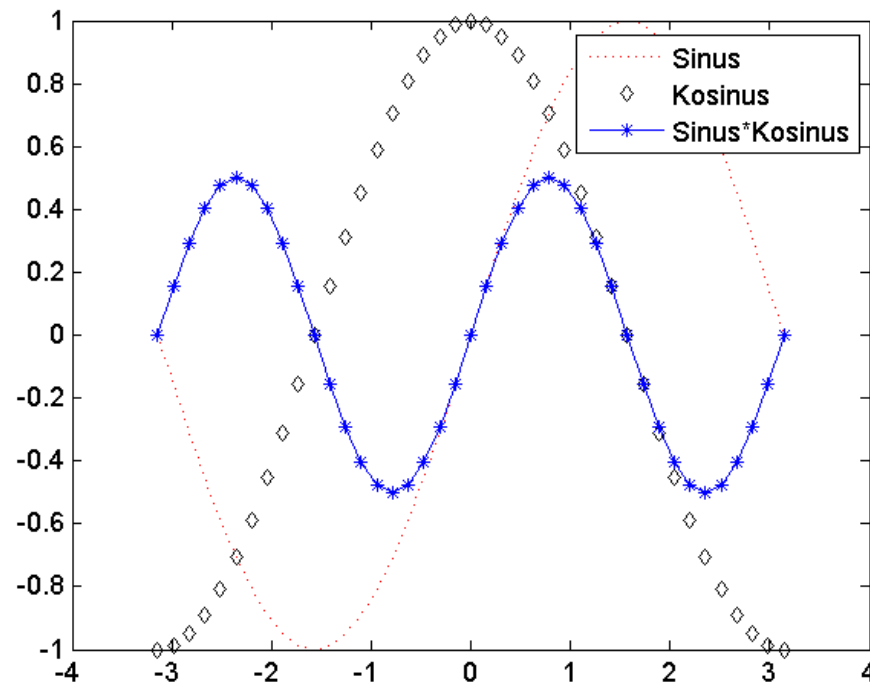
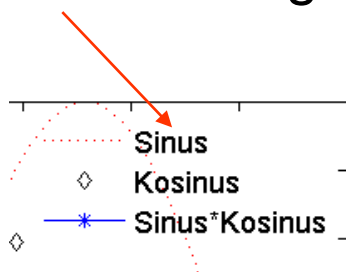
Postavljanje legende na grafik

- `legend('tekst1', 'tekst2', 'tekst3', ...)` – postavljanje legende na grafik (po redosledu navođenja u `plot` funkciji).

- Primer:

```
x = linspace(-pi,pi,41);  
y1 = sin(x); y2 = cos(x); y3 = y1.*y2;  
plot(x,y1,'r:',x,y2,'kd',x,y3,'b-*')  
legend('Sinus', 'Kosinus', 'Sinus*Kosinus')
```

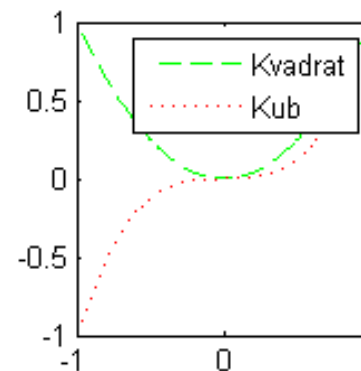
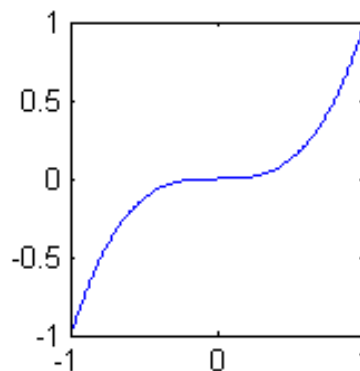
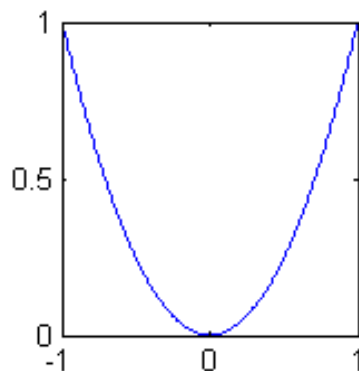
- Naredbom `legend boxoff` se uklanja okvir oko legende.



Dijeljenje grafika

- **figure** – otvara novi grafički prozor
- **subplot(x,y,z)** – dijeli grafički prozor na x djelova po vertikali, y po horizontali i pozicionira se u dijelu z za crtanje narednog grafika.

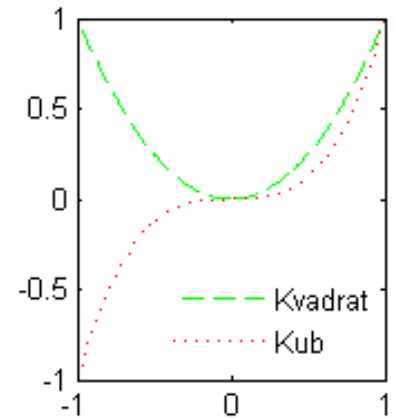
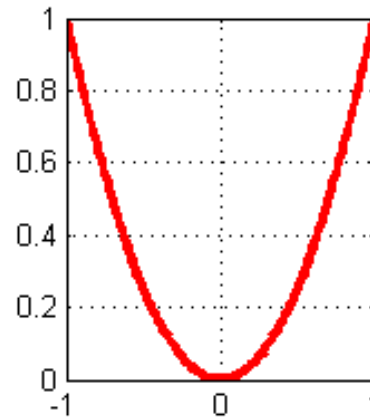
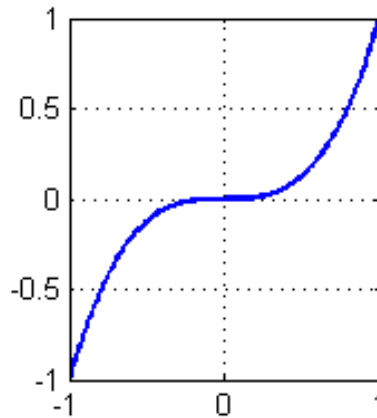
```
>> x = -1 : 1/99 : 1;  
>> y1 = x .^ 2;  
>> y2 = x .^ 3;  
>> subplot(1, 3, 1)  
>> plot(x, y1)  
>> subplot(1, 3, 2)  
>> plot(x, y2)  
>> subplot(1, 3, 3)  
>> plot(x, y1, '--g', x, y2, 'r:')  
>> legend('Kvadrat', 'Kub')
```



Postavljanje legende i dijeljenje grafika

■ `grid` – postavljanje mreže

```
>> x = -1:1/99:1;
>> y1 = x.^2;
>> y2 = x.^3;
>> subplot(1,3,1)
>> plot(x,y2, 'linewidth',2); grid
>> subplot(1,3,2)
>> plot(x,y1, 'color','r', 'linewidth',3); grid
>> subplot(1,3,3)
>> plot(x,y1, 'g--',x,y2, 'r:');
>> legend('Kvadrat', 'Kub', 'location','best')
>> legend boxoff
```



Polarne koordinate

- **polar(fi,ro)** – crta funkciju u polarnom koordinatnom sistemu, funkciju $\rho = \rho(\varphi)$. Ovdje je φ ugao a ρ udaljenost od centra
- **linspace(a,b,N)** – definisanje vektora od **N** tačaka u intervalu **[a,b]**
- Primjer: nacrtati grafik kardioide $r = a(1 + \cos\varphi)$, $-\pi \leq \varphi \leq \pi$

```
>> phi=linspace(-pi,pi,50);
```

```
>> r=2*(1+cos(phi));
```

```
>> polar(phi,r,'r*')
```

