

# **NİĞDE ÜNİVERSİTESİ**

Elektrik Elektronik Mühendisliği Bölümü

Devre Tasarımı Ders Notları

## **MATLAB**

Arş. Gör. Salim ÇINAR

## MATLAB ÖRNEKLERİ

Atamalar:

```
>> a=5
```

```
a =
```

```
5
```

```
>>
```

Çıkan sonucun görünmesi istenmiyorsa atamadan sonra “;” konulur.

```
>> bb=28;
```

```
>>
```

İşlemler:

```
>> 5+6
```

```
ans =
```

```
11
```

```
>> 2*29
```

```
ans =
```

```
58
```

```
>> 80/4
```

```
ans =
```

```
20
```

```
>> 34-38
```

```
ans =
```

```
-4
```

```
>>
```

```
>> 4^5      Üs alma
```

```
ans =
```

```
1024
```

```
>>
```

```
>> sqrt(256)      Karekök alma
```

```
ans =
```

```
16
```

```
>>
```

```
>> pi
```

```
ans =
```

```
3.1416
```

```
>>
```

Trigonometrik fonksiyonlardan bazıları aşağıda verilmiştir.

Sin(x), sinh(x), asin(x), asinh(x), cos(x), cosh(x), acos(x), acosh(x), tan(x), tanh(x), atan(x), atanh(x) vs...

```
>> sin(30)
```

```
ans =
```

```
-0.98803
```

```
>> sin(pi*30/180)
```

```
ans =
```

```
0.5
```

```
>>
```

```
>> exp(2) % e2
```

```
ans =
```

```
7.3891
```

```
>>
```

```
>> exp(1)
```

```
ans =
```

```
2.7183
```

```
>>
```

```
>> log(10) % loge10
```

```
ans =
```

```
2.3026
```

```
>>
```

```
>> log10(5) % log105
```

```
ans =
```

```
0.69897
```

```
>>
```

```
>> log2(8) % log28
```

```
ans =
```

```
3
```

```
>>
```

İşlem önceliği: Parantez içi, üs alma, çarpma/bölme, toplama/çıkarma. Çarpma ve bölmeler art arda ise işlemler soldan sağa doğru yapılır. Aynı işlemler toplama çıkarma içinde geçerlidir.

```
>> 3*4/2+23-16
```

```
ans =
```

```
13
```

```
>>
```

```
>> 4*(3+3^3)/5^2+22/3
```

$$4x \frac{(3+3^3)}{5^2} + \frac{22}{3}$$

```
ans =
```

```
12.133
```

```
>>
```

Vektör ve Matrisler

```
>> a=[1 2 3 4 5 6]
```

```
a =
```

```
1 2 3 4 5 6
```

```
>>
```

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

```
a =
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
>>
```

```
>> b=[2 3 4;1 1 4;6 -12 8]
```

```
b =
```

```
    2    3    4  
    1    1    4  
    6   -12    8
```

```
>>
```

```
>> cc=[-1:0.5:1]
```

```
cc =
```

```
   -1   -0.5    0    0.5    1
```

```
>>
```

```
>> a=zeros(3,4)           % Tamamı sıfırlardan oluşan matris. 3 satır x 4 sütun
```

```
a =
```

```
    0    0    0    0  
    0    0    0    0  
    0    0    0    0
```

```
>>
```

```
>> b=ones(5,3)
```

```
b =
```

```
    1    1    1  
    1    1    1  
    1    1    1  
    1    1    1  
    1    1    1
```

```
>>
```

```
>> size(a)          % a matrisinin boyutunu verir.
```

```
ans =
```

```
    3    4          % satır sayısı=3, sütun sayısı=4.
```

```
>>
```

```
>> whos           % bu kod Workspace'de kullanılan değişkenler hakkında bilgi verir.
```

```
 Name   Size      Bytes Class  Attributes
```

```
 a      3x4         96   double
```

```
 ans    1x2         16   double
```

```
 b      5x3        120   double
```

```
>>
```

```
>> b=[2 3 4;1 1 4;6 -12 8]
```

```
b =
```

```
    2    3    4
```

```
    1    1    4
```

```
    6   -12    8
```

```
>> b(1,2)         % tanımlanan matrisin istenilen satır ve sütündeki elemanına ulaşmak için  
                  %kullanılır. İlk parametre satırı, ikinci parametre ise sütunu temsil etmektedir.
```

```
ans =
```

```
    3
```

```
>> b(3,2)
```

```
ans =
```

```
   -12
```

```
>> c=b(1,:)      % Matrisin istenilen satırını almak için kullanılır.
```

```
c =
```

```
    2    3    4
```

```
>> d=b(:,3) % Matrisin istenilen sütununu almak için kullanılır.
```

```
d =
```

```
4  
4  
8
```

```
>>
```

```
>> e=b(1:3,2:3) % Matrisin istenilen bölgesini almak için kullanılan komut
```

```
e =
```

```
3 4  
1 4  
-12 8
```

```
>> b(2,2)=9 % Matrisin istenilen elemanını değiştirmek için kullanılan komut.
```

```
b =
```

```
2 3 4  
1 9 4  
6 -12 8
```

```
>> b(1,3)=7
```

```
b =
```

```
2 3 7  
1 9 4  
6 -12 8
```

```
>> b(2:3,1:3)=[3 2 -5;11 3 15]
```

```
b =
```

```
2 3 7  
3 2 -5  
11 3 15
```



```
>> rand(3,2)          % 3x2 lik 0-1 arası rastgele matris oluşturmak.
```

```
ans =
```

```
    0.81472    0.91338  
    0.90579    0.63236  
    0.12699    0.09754
```

```
>> a=ceil(19*rand(3,3))    % 0-19 arası 3x3 lük rastgele matris üretme kodu
```

```
a =
```

```
    15    14    11  
     5    17     3  
    10    19     3
```

```
>> b=ceil(19*rand(3,3))
```

```
b =
```

```
     5    16     7  
    16     5     4  
     5    18     5
```

```
>> c=a+b          % İki matrisi her bir elemanı kendi arasında toplanarak c matrisine  
                  % atanmıştır.
```

```
c =
```

```
    20    30    18  
    21    22     7  
    15    37     8
```

```
>> d=a-b
```

```
d =
```

```
    10    -2     4  
   -11    12    -1  
     5     1    -2
```

```
>>
```

```
>> kk=ceil(4*rand(2,2))
```

```
kk =
```

```
3 2
2 4
```

```
>> nn=ceil(4*rand(2,2))
```

```
nn =
```

```
3 4
3 2
```

```
>> mm=kk*nn % Matris çarpımı Matrislerin boyutu MxN ye NxK
```

```
mm =
```

```
15 16
18 16
```

```
>> kk=ceil(4*rand(3,3))
```

```
kk =
```

```
1 4 1
2 2 3
1 3 2
```

```
>> nn=ceil(5*rand(3,3))
```

```
nn =
```

```
3 3 1
5 1 4
1 5 5
```

```
>> mm=kk.*nn % Burada her bir matris elemanı karşısındaki elemanla çarpılmıştır. Her  
%iki matrisin boyutları eşit olmalıdır.
```

```
mm =
```

```
3 12 1
10 2 12
1 15 10
```

```
>> aa=ceil(19*rand(3,3))
```

```
aa =
```

```
17  5 18
 2 16  4
 8  9  6
```

```
>> bb=aa'    % aa nın tranpozesini almak için kullanılır.
```

```
bb =
```

```
17  2  8
 5 16  9
18  4  6
```

```
>> cc=inv(aa)    % aa nın tersini almak için kullanılır.
```

```
cc =
```

```
-0.069767 -0.15349  0.31163
-0.023256 0.048837  0.037209
 0.12791  0.1314 -0.30465
```

```
>> dd=det(aa)    % aa nın determinantını hesaplamak için kullanılan komut.
```

```
dd =
```

```
-860
```

```
>>
```

```
>> rank(aa)    % aa matrisinin rankını bulmak için kullanılan komut.
```

```
ans =
```

```
3
```

```
>>
```

```
>> ee=2*aa          % aa matrisinin tüm elemanları belli bir katsayı ile çarpmak için gereken
                    %kod
```

```
ee =
```

```
34  10  36
 4   32   8
16  18  12
```

```
>> ff=aa.^3        % aa matrisinin her bir elemanının 3. Kuvveti alınmıştır.
```

```
ff =
```

```
4913   125   5832
    8   4096    64
 512   729   216
```

```
>> gg=aa^3        % aa matrisinin küpünün alınması için gerekli kod. ( aa*aa*aa ).
```

```
gg =
```

```
11657   11353  11886
 3262   6438   3716
 5638   6762   5884
```

```
>> trace(aa)       %Matrisin diagonal elemanlarının toplamını verir.
```

```
ans =
```

```
39
```

Dizilerde yapılan işlemler:

```
>> vv=ceil(100*rand(1,14))
```

```
vv =
```

```
12  79  39  25  41  10  14  95  96  58  6  24  36  83
```

```
>> max(vv)         % Dizinin en büyük elemanını bulur.
```

```
ans =
```

```
96
```

```
>> min(vv)          % Dizinin en küçük elemanını bulur.

ans =

    6

>> mean(vv)        %Dizinin ortalamasını bulduran komut.

ans =

    44.143

>> median(vv)

ans =

    37.5

>> std(vv)         % Dizinin standart sapmasını bulur.

ans =

    32.38

>> sort(vv)        % Diziyi küçükten büyüğe doğru sıralar.

ans =

    6  10  12  14  24  25  36  39  41  58  79  83  95  96

>> sum(vv)         % Dizideki tüm elemanları toplar.

ans =

    618
```

## Fonksiyon ve M file kullanımı:

Fonksiyonlar programlama dillerinde olduğu gibi Matlab’de de kullanılır. Yeni bir fonksiyon tanımlama için File>>New>>Blank M-file veya Function M-file seçilir. Fonksiyonun yazımı şu şekildedir.

```
function [ out ] = topla( tp1,tp2 )  
  
% Bu fonksiyon tp1 ve tp2 nin toplamını out değişkenine atar.  
out=tp1+tp2;  
  
end
```

Burada out fonksiyonun geriye döndüreceği değeri gösterir. topla ise fonksiyonun ismidir. tp1 ve tp2 ise fonksiyonun giriş parametreleridir. Fonksiyonu yazdıktan sonra M-file fonksiyon\_ismi.m şeklinde kaydedilir. Bu örnekte fonksiyon topla.m olarak kaydedilmiştir. Fonksiyonun kullanımı şu şekildedir.

```
>> gg=topla(23.17,54.98)
```

```
gg =
```

```
    78.15
```

```
>>
```

## Karar Yapıları ve Döngüler:

if : Komutları belirlenen şarta göre işlemek için kullanılır.

```
IF karşılaştırma  
  komutlar  
ELSEIF karşılaştırma  
  komutlar  
ELSE  
  komutlar  
END
```

==, <, >, <=, >=, ~= ( Karşılaştırma durumları)

Örnek:

```
if I == J  
  A(I,J) = 2;  
elseif abs(I-J) == 1  
  A(I,J) = -1;  
else  
  A(I,J) = 0;  
End
```

Switch: değişkenin aldığı değerlere göre komutları işleyen karar yapısıdır.

```
SWITCH deger
  CASE durum1,
    komutlar
  CASE {durum2, durum3, durum4,...}
    komutlar
...
  OTHERWISE,
    komutlar
END
```

```
a=input('Bir rakam giriniz= ');
switch a
  case 0,
    display('Sıfır');
  case 1,
    display('Bir');
  case 2,
    display('İki');
  case 3,
    display('Üç');
  case 4,
    display('Dört');
  case 5,
    display('Beş');
  case 6,
    display('Altı');
  case 7,
    display('Yedi');
  case 8,
    display('Sekiz');
  case 9,
    display('Dokuz');
  otherwise
    display('Rakam giriniz!');
end
```

Yukarıdaki örnek “case\_deneme.m” isimli m-file olarak kaydedilmiştir. Komut penceresine case\_deneme yazıldığında ekrana gelen kod aşağıda verilmiştir.

```
>> case_deneme
Bir rakam giriniz= 3
Üç
>> case_deneme
Bir rakam giriniz= 10
Rakam giriniz!
>>
```

for: belirtilen sayı kadar döngüde işlemleri gerçekleştir.

```
>> tt=0;  
>> for i=1:100  
tt=tt+i;  
end  
>> tt
```

tt =

5050

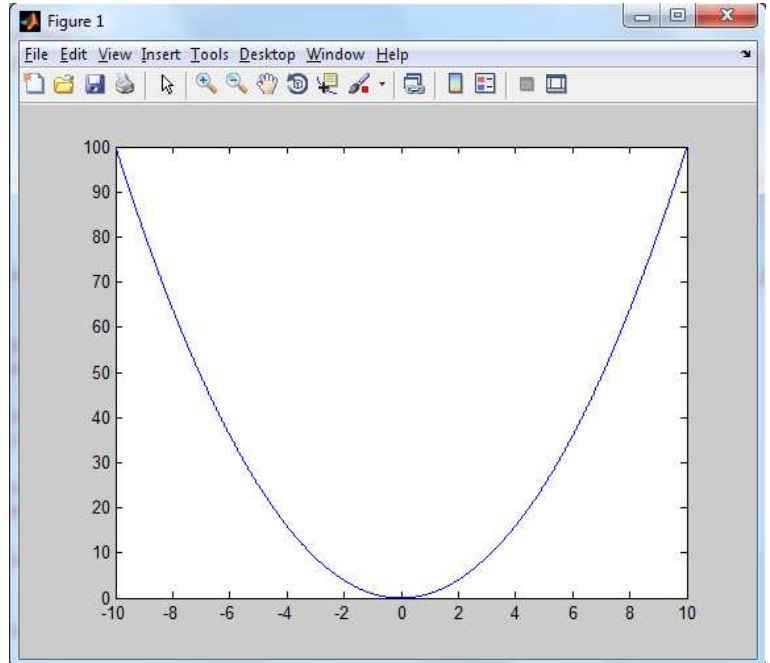
While: belirtilen şart sağlanıncaya kadar döngüye devam edilir.

```
>> hh=0;  
>> while (hh<1000)  
hh=hh+1;  
end  
>> hh  
hh =
```

1000

### Temel Grafik Komutları

```
>> x=[-10:0.01:10];  
>> y=x.^2;  
>> plot(x,y);
```

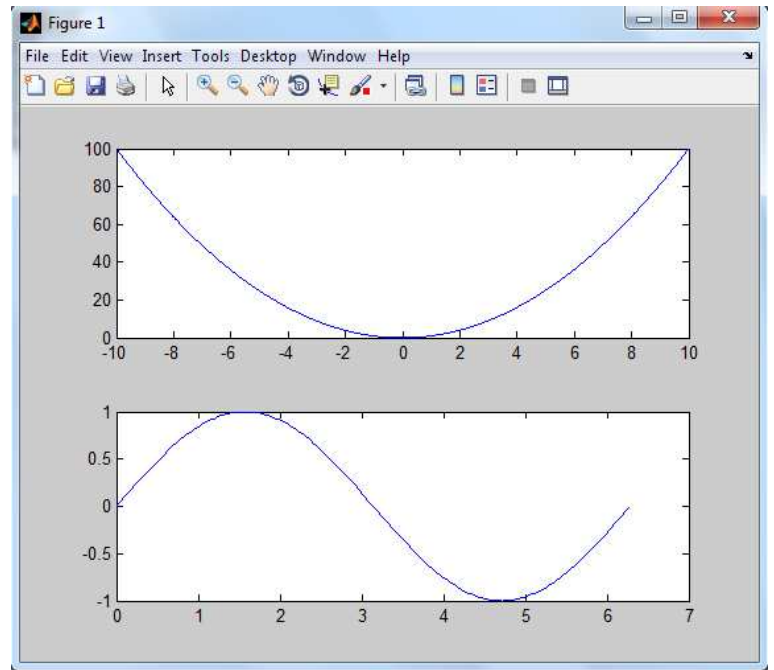




```

>> x=[-10:0.01:10];
>> y=x.^2;
>> subplot(2,1,1)
>> plot(x,y);
>> n=[0:pi/100:2*pi];
>> m=sin(n);
>> subplot(2,1,2)
>> plot(n,m);

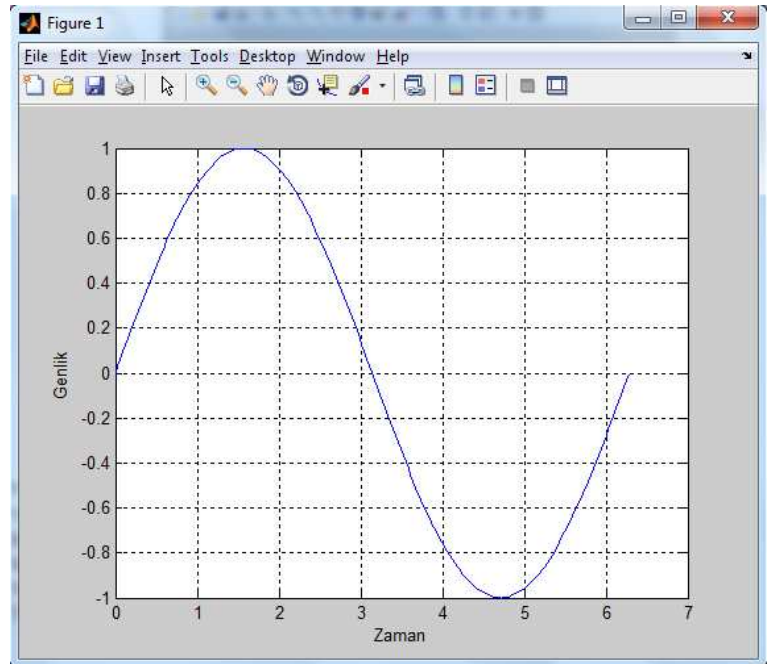
```



```

>> t = 0:pi/100:2*pi;
>> y = sin(t);
>> plot(t,y)
>> grid on % Grid göster
>> xlabel('Zaman');
>> ylabel('Genlik');

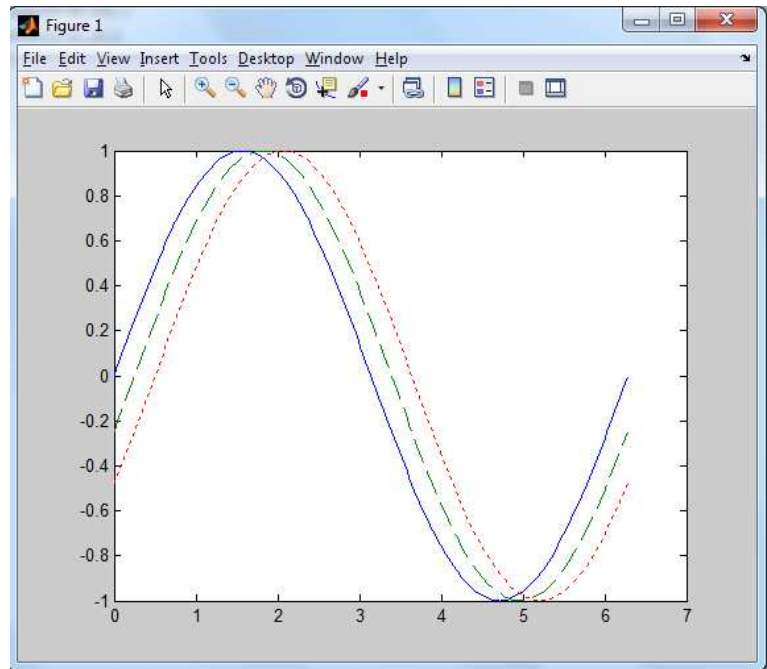
```



```

t = 0:pi/100:2*pi;
y = sin(t);
y2 = sin(t-0.25);
y3 = sin(t-0.5);
plot(t,y,'-',t,y2,'--',t,y3,':')

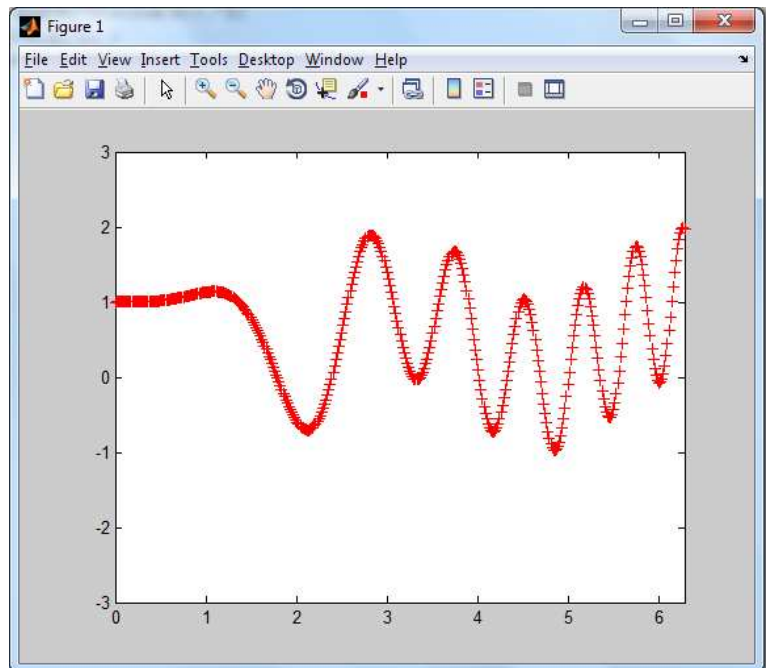
```



```

>> t = 0:pi/200:2*pi;
>> y = sin(t.^2)+(cos(t)).^2;
>> plot(t,y,'r+')
>> axis([0 2*pi -3 3])

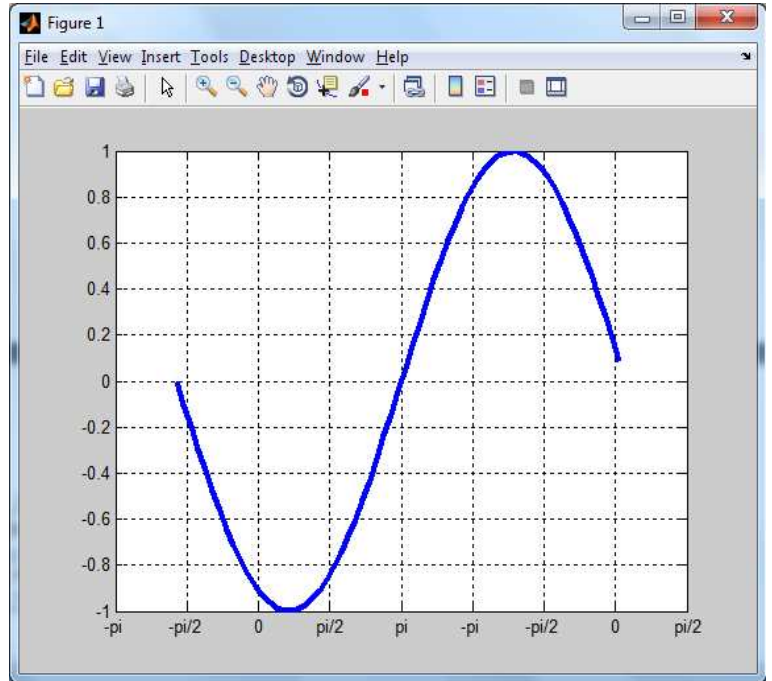
```



```

>> x = -pi:1:pi;
>> y = sin(x);
>> plot(x,y,'linewidth',3)
>> grid on % Grid göster
>> set(gca,'XTickLabel',{'-pi','-pi/2','0','pi/2','pi'})

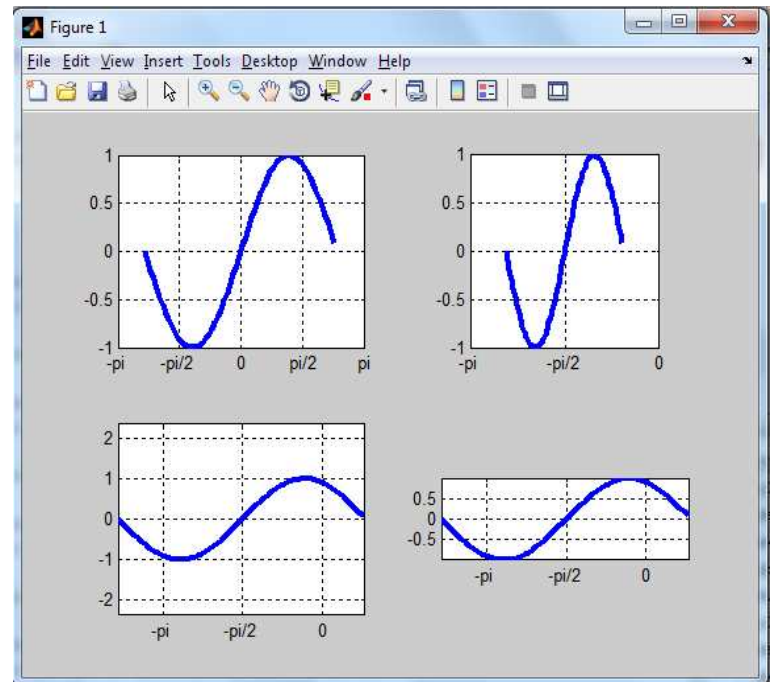
```



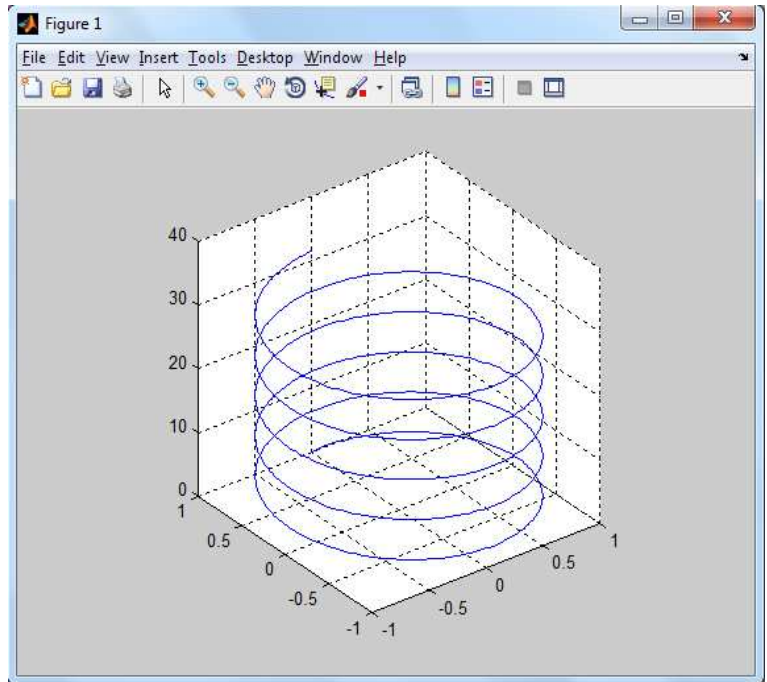
```

>> x = -pi:1:pi;
>> y = sin(x);
>> subplot(2,2,1)
>> plot(x,y,'linewidth',3)
>> set(gca,'XTickLabel',{'-pi','-pi/2','0','pi/2','pi'})
>> axis normal
>> grid on % Grid göster
>> subplot(2,2,2)
>> plot(x,y,'linewidth',3)
>> set(gca,'XTickLabel',{'-pi','-pi/2','0','pi/2','pi'})
>> axis square
>> grid on % Grid göster
>> subplot(2,2,3)
>> plot(x,y,'linewidth',3)
>> set(gca,'XTickLabel',{'-pi','-pi/2','0','pi/2','pi'})
>> axis equal
>> grid on % Grid göster
>> subplot(2,2,4)
>> plot(x,y,'linewidth',3)
>> set(gca,'XTickLabel',{'-pi','-pi/2','0','pi/2','pi'})
>> axis equal tight
>> grid on % Grid göster

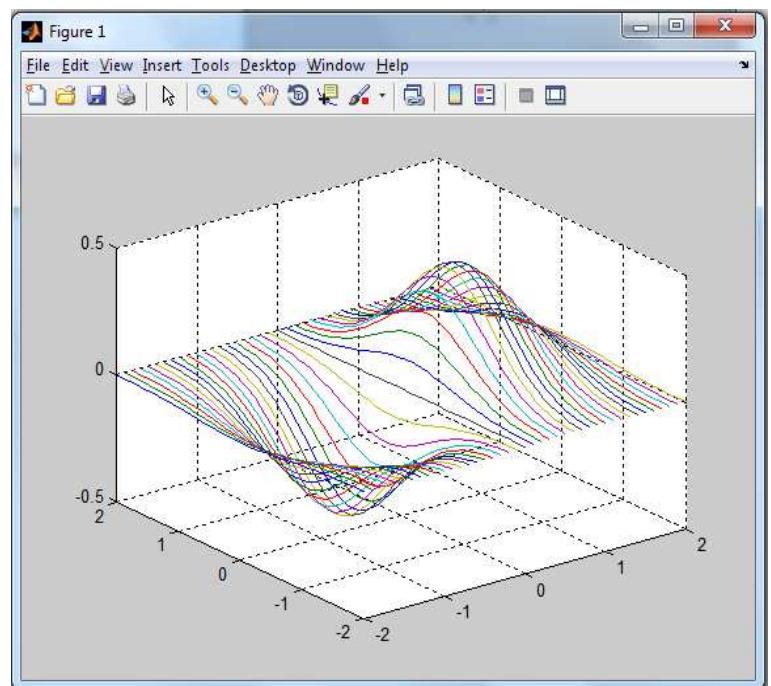
```



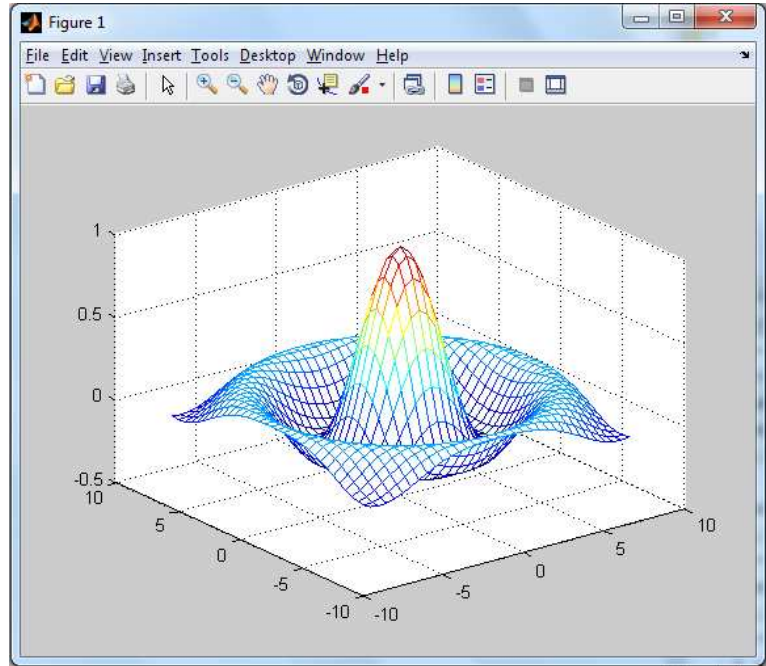
```
>> t = 0:pi/100:10*pi;  
plot3(sin(t),cos(t),t)  
axis square; grid on
```



```
[X,Y] = meshgrid([-2:0.1:2]);  
Z = X.*exp(-X.^2-Y.^2);  
plot3(X,Y,Z)  
grid on
```

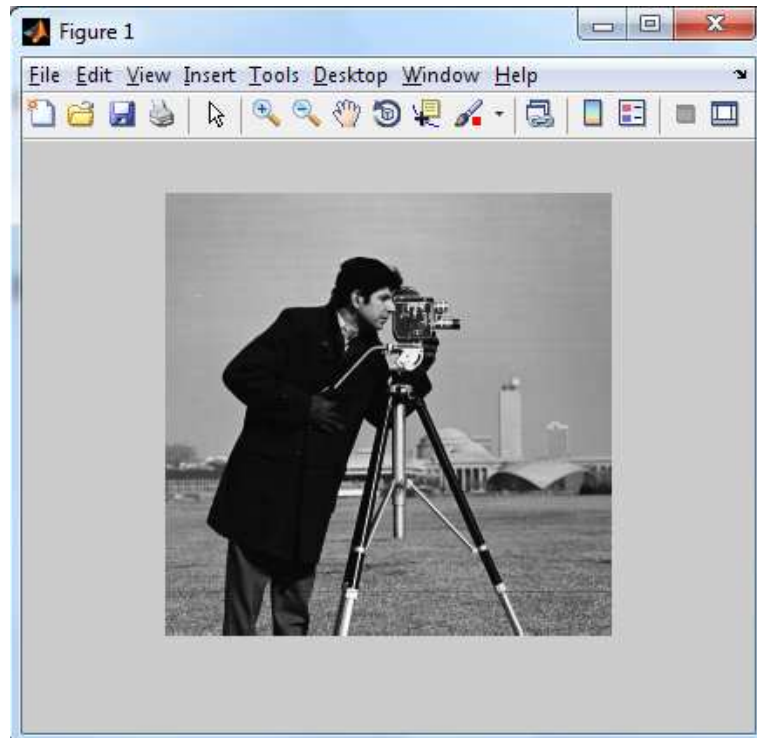


```
>> [X,Y] = meshgrid(-8:5:8);  
>> R = sqrt(X.^2 + Y.^2) + eps;  
>> Z = sin(R)./R;  
>> mesh(X,Y,Z)
```

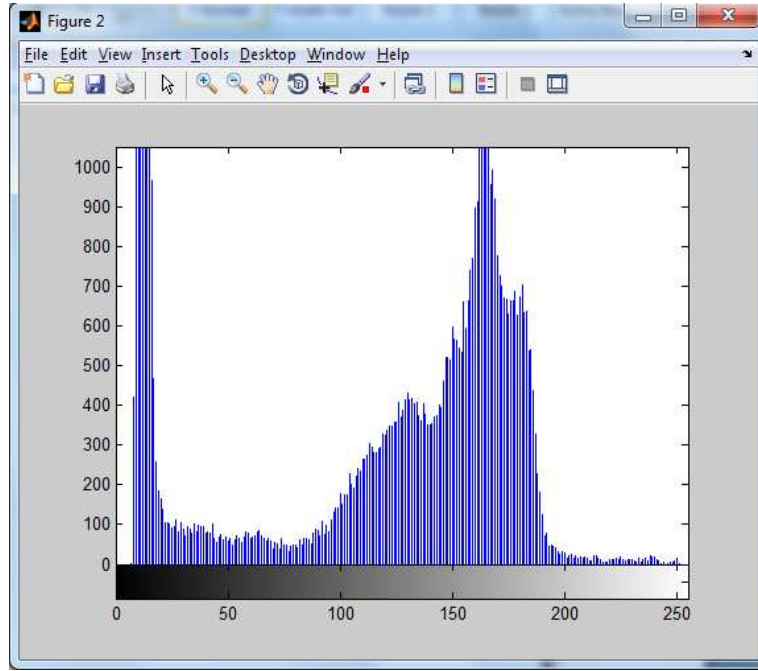


## Temel Resim İşleme

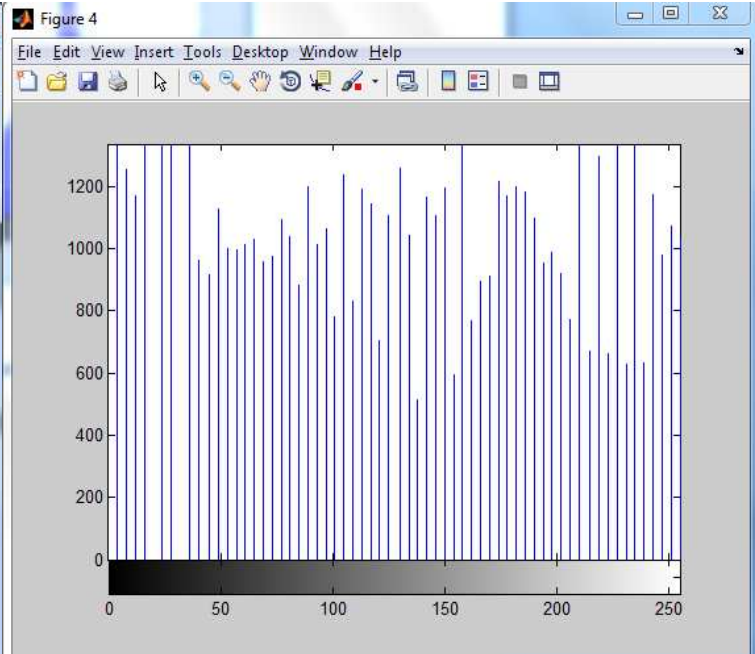
```
>> resim=imread('cameraman.tif'); % Resimi dosyadan okuyor  
>> whos  
Name      Size      Bytes Class  Attributes  
  
resim     256x256    65536 uint8  
  
>> imshow(resim); % Resimi Figure penceresinde gösteriyor
```



>> figure, imhist(resim); % Resmin histogramını çıkartmak için kullanılan kod.



>> resim2 = histeq(resim); % burada histogram eşitleme işlemi yapılıyor.  
>> figure, imshow(resim2);  
>> figure, imhist(resim2);  
>> imwrite(resim2, 'cameraman2.tif'); % Bu komutla yeni oluşan görüntü kaydediliyor.



Örnek: Aşağıda yapay sinir ağı kullanılarak çarpım tablosu oluşturulmuştur.

```
>> egitim=round(10*rand(2,200));  
>> for i=1:200  
hedef(1,i)=egitim(1,i)*egitim(2,i);  
end  
>> net=newfit(egitim,hedef,[5 15 5 1]);  
>> [net,tr]=train(net,egitim,hedef);  
>> ann=[10;3];  
>> snc=sim(net,ann)
```

snc =

29.9999650289685

```
>> ann=[5;9];  
>> snc=sim(net,ann)
```

snc =

44.9927472707975

```
>>
```