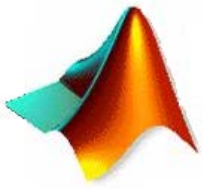


# SAYISAL ANALİZ

**Doç.Dr. Cüneyt BAYILMIŞ**





# SAYISAL ANALİZ

## 2. Hafta

### MATLAB İLE PROGRAMLAMA

# İÇİNDEKİLER

## 1. .m dosya kullanımı

- ☐ **script** tanımlaması
- ☐ **function** tanımlaması

## 2. Programlamanın Temelleri

- ☐ Klavyeden okuma ve ekrana yazdırma
- ☐ Değişken tanımlamanın kuralları
- ☐ Aritmetik/Mantıksal İşleçler

## 3. Karşılaştırma ve Karar Verme Yapıları

- ☐ **if/else**
- ☐ **switch/case**

## 4. Döngü Yapıları

- ☐ **for**
- ☐ **while**

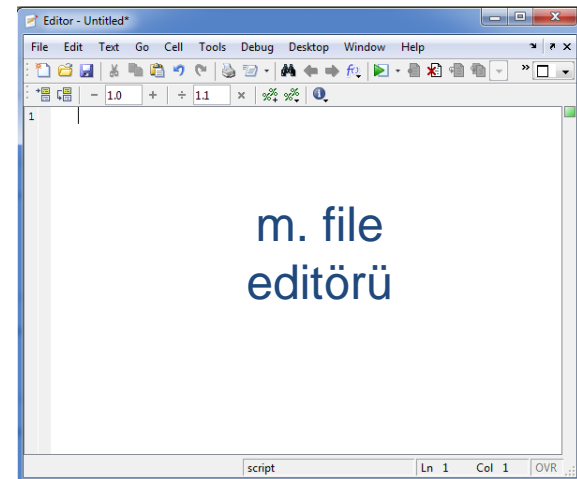
## 5. Örnek Uygulamalar



# MATLAB İLE PROGRAMLAMA

- ❑ MATLAB'ta tek satır ya da birkaç satırdan oluşan işlemleri komut penceresinde kolaylıkla yapılabilir.
- ❑ Uzun işlemleri yapmak ise zordur.
- ❑ Bunun için programlama mantığı içerisinde kodların bir dosya (.m file) içerisinde yazılarak çalıştırılması tercih edilir.
- ❑ Program (.m file) dosyasını çalıştırmak için komut penceresinde dosyanın adını yazmak yeterlidir. Programı oluşturan tüm satırlar çalıştırılır.
- ❑ .m dosyaları işleyiş şekline göre;

- ① Script
- ② Fonksiyon



# MATLAB'ta Genel Tanımlama

## ❑ Değişken tanımlarken uyulması gereken kurallar

- 1 İngiliz alfabesindeki A-Z veya a-z arası 26 harf kullanılabilir
- 2 0-9 arası rakamlar kullanılabilir
- 3 Simgelerden sadece alt çizgi ( \_ ) kullanılabilir
- 4 Tanımlayıcı isimleri, harf veya alt çizgi ile başlayabilir
- 5 Rakam ile başlayamaz veya sadece rakamlardan oluşamaz
- 6 Kullanılan programlama dilinin komutu ya da saklı kelimelerinden olamaz
- 7 Büyük – Küçük harf duyarlıdır. **A** ile **a** farklı değişkenleri ifade eder.

# MATLAB'ta Genel Tanımlama

- ❑ MATLAB'ta her şey bir dizi (array) olarak işlenir.
- ❑ Dizi;
  - ❑ Tek bir değerden oluşuyorsa (1x1) **skaler** olarak adlandırılır.
    - **Örnek:**  $a=5$  ,  $b=-17.33$ ,  $c=2e-3$ ,  $d=3+4*i$
  - ❑ Tek bir satır ya da sütunda bir den fazla değerden oluşuyorsa (**1xn ya da nx1 dizisi şeklinde ise**) **vektör** olarak adlandırılır.
    - **Örnek:**  $a=[1\ 3\ 5\ 7]$
  - ❑ Birden fazla satır ve sütundan oluşuyorsa (**nxm ya da mxn dizisi şeklinde ise**) **matris** olarak adlandırılır.
    - **Örnek:**  $a=[1\ 3\ 5\ 7; 2\ 4\ 6\ 8]$
    - Satırlar arasındaki ayırım **;** ile yapılır.

# Klavyeden Bilgi Girme ve Ekrana Bilgi Yazma

## ❑ Bilgi girişi

### ❑ input

- **Sayısal değer girme:** `değişken=input('sayi gir');`
- **Karakter girme:** `değişken=input('karakter gir', 's');`

## ❑ Ekrana yazdırma

### ❑ disp

- **Mesaj yazdırma:** `disp('açıklama');`
- **Değişken yazdırma:** `disp(değişken);`

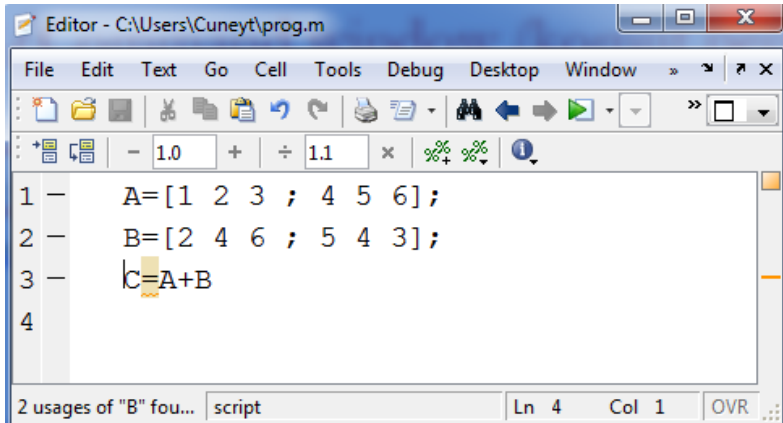
### ❑ fprintf

- **Mesaj yazdırma:** `fprintf('açıklama');`
- **Değişken yazdırma:** `fprintf('açıklama = % f', değişken);`

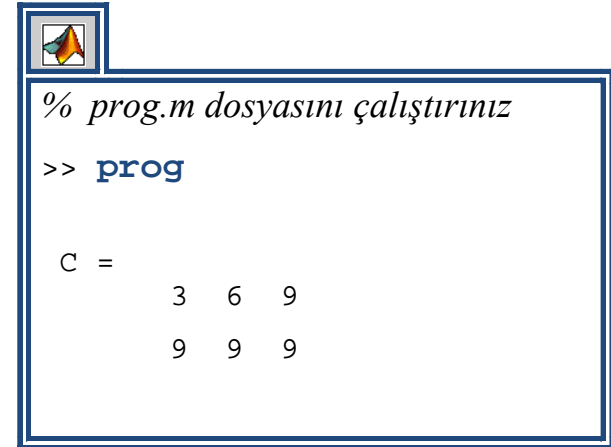
- **% f** : ondalıklı sayı                      **% g** : tam sayı
- **% d** : işaretli ondalıklı sayı   **% s** : karakter
- **\n** bir alt satıra geç                      **\t** bir tab boşluk

# ÖRNEKLER

- 1 A ve B şeklinde 2 matris tanımlayarak toplayan programı yazınız?



```
Editor - C:\Users\Cuneyt\prog.m
File Edit Text Go Cell Tools Debug Desktop Window
1 - A=[1 2 3 ; 4 5 6];
2 - B=[2 4 6 ; 5 4 3];
3 - C=A+B
4
2 usages of "B" fou... script Ln 4 Col 1 OVR
```

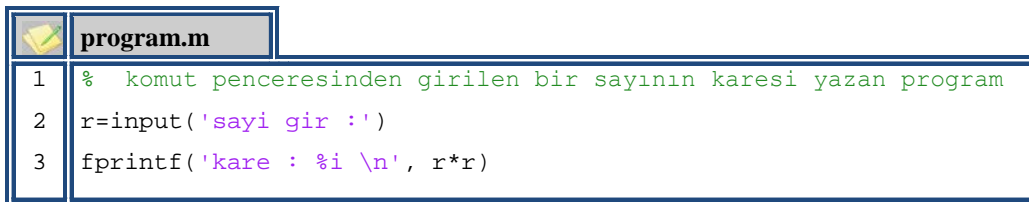


```
% prog.m dosyasını çalıştırınız
>> prog

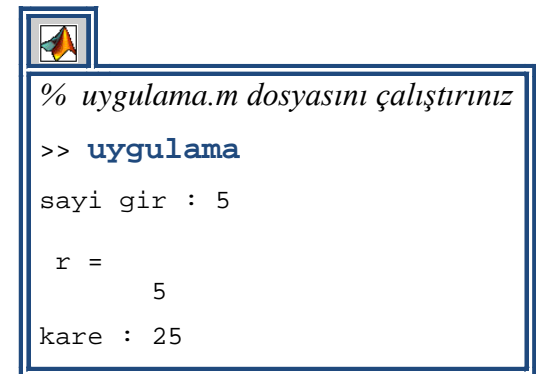
C =

     3     6     9
     9     9     9
```

- 2 Komut penceresinden girilen bir sayının karesi hesaplayarak ekrana yazdıran programı yazınız?



```
program.m
1 % komut penceresinden girilen bir sayının karesi yazan program
2 r=input('sayi gir :')
3 fprintf('kare : %i \n', r*r)
```



```
% uygulama.m dosyasını çalıştırınız
>> uygulama

sayi gir : 5

r =

     5

kare : 25
```



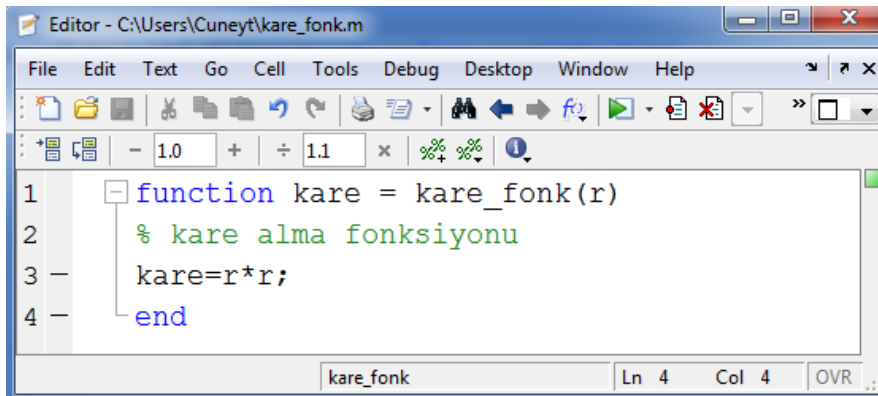
# Fonksiyon Oluşturma

## ❑ Fonksiyonun genel formatı

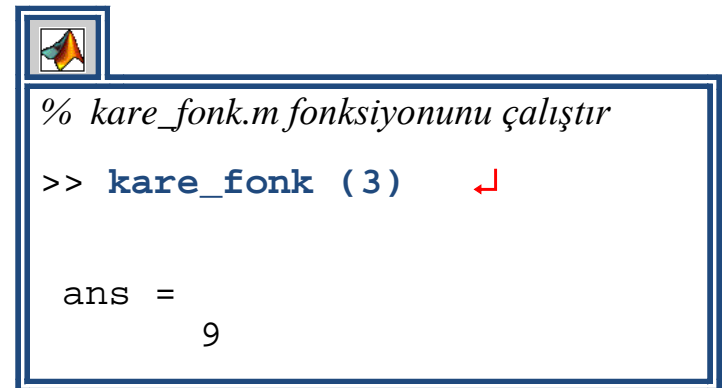
	Fonksiyon_Adi.m
1	<code>function GeriDönüşDeğeri = Fonksiyon_Adi (Giriş Parametreleri)</code>
2	<code>% Function satırını takip eden ilk açıklama satırları ilgili</code>
3	<code>% fonksiyonun yardım metinlerini oluşturur</code>
4	<code>işlemler;</code>
5	<code>end</code>

Not: Fonksiyon ismi ile fonksiyon dosya adı aynı olmalıdır.

## ③ Girilen sayının karesini hesaplayarak ekrana yazdıran programı fonksiyon olarak yazınız?



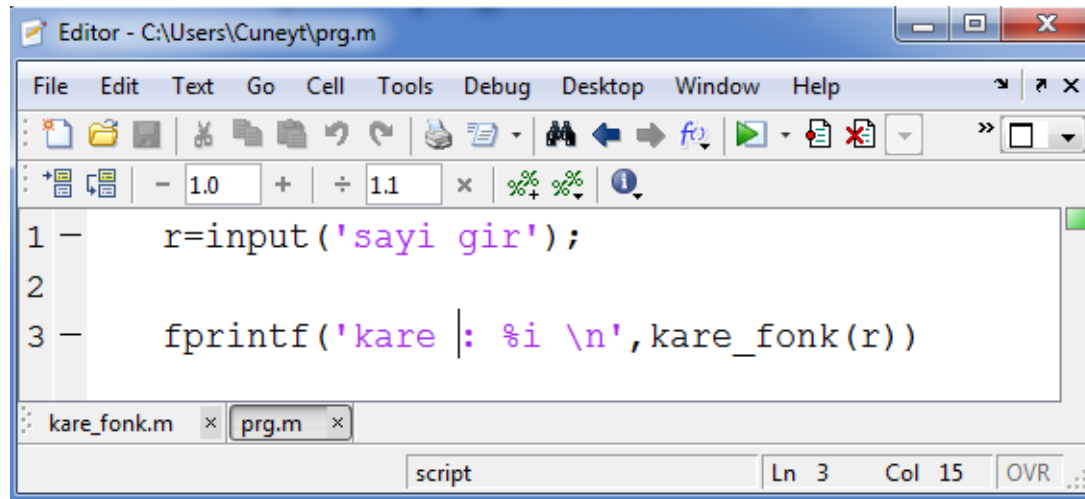
```
1 function kare = kare_fonk(r)
2 % kare alma fonksiyonu
3 kare=r*r;
4 end
```



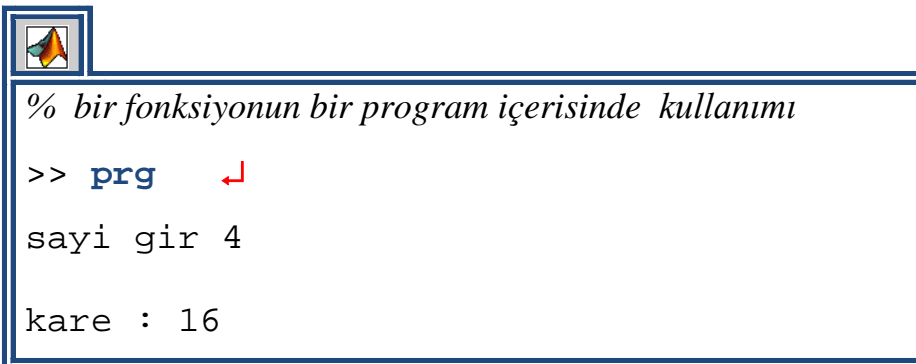
```
% kare_fonk.m fonksiyonunu çalıştır
>> kare_fonk (3)
ans =
     9
```

# Bir Fonksiyonun Başka Bir Script İçerisinde Kullanımı

- 4 Ana programda klavyeden sayı girilmesi istenecek, girilen sayının karesi önceki soruda tanımlanan **fonksiyon** yardımıyla hesaplanarak ana programa geri döndürülüp, ana program ile ekrana yazdıran programı yazınız



```
1 - r=input('sayi gir');
2
3 - fprintf('kare : %i \n',kare_fonk(r))
```



```
>> prg
sayi gir 4
kare : 16
```

# Aritmetik İşleçler

İşleç	Açıklama	İşleç	Açıklama
+	Toplama	'	Transpoze
-	Çıkartma	.*	Dizi Çarpma (Eleman-Eleman)
.	Ondalık Noktası	./	Dizi Sağa Bölme (Eleman-Eleman)
=	Atama	.\	Dizi Sola Bölme (Eleman-Eleman)
*	Çarpma	.^	Dizi Üs Alma (Eleman-Eleman)
/	Sağa Bölme	.'	Dizi Tranpoze
\	Sola Bölme	^	Üs Alma

# Matematiksel İşleçlerde Öncelik

Sıra	İşlem	Bilgisayar
1	Sayıların Negatifliği	-...
2	Parantezler (içten dışa doğru)	( .... )
3	Matematiksel Fonksiyonlar	cos, sin, log, ...
4	Üs alma	$a^b$ ,
5	Çarpma ve Bölme (soldan sağa)	$a * b$ ve $a/b$
6	Toplama ve Çıkarma (soldan sağa)	$a + b$ ve $a - b$

# Matematiksel İşlemler

## ➤ Örnek:

### ❑ Matematiksel ifade :

$$x = a \cdot b / c + d \cdot e^f - g$$

### ❑ Bilgisayar ifadesi:

$$x = a \quad * \quad b \quad / \quad c \quad + \quad d \quad * \quad e \wedge f - g$$

**2        3        4        2        1        5**

# Karşılaştırma İşleçler

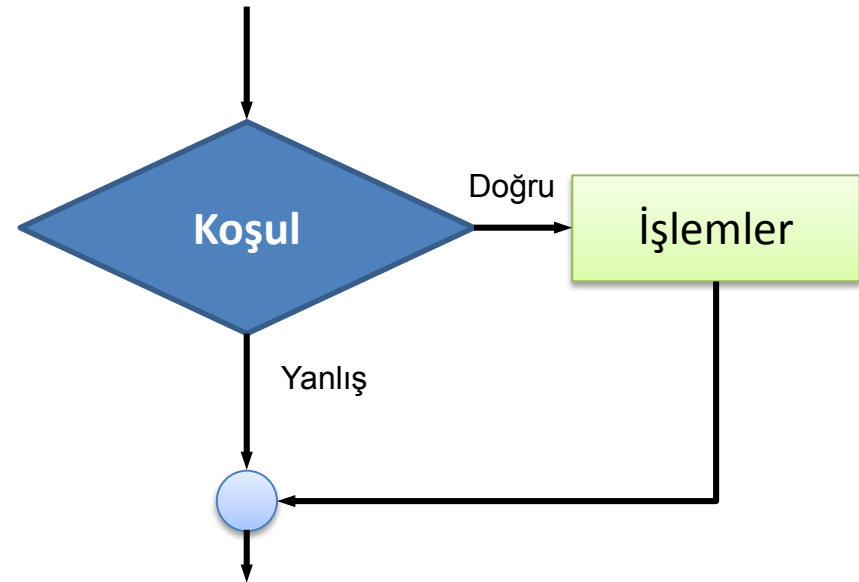
Sembol	Anlamı
$=$	Eşittir
$\approx$	Eşit Değildir
$>$	Büyüktür
$<$	Küçüktür
$\geq$	Büyük eşittir
$\leq$	Küçük eşittir

# Mantıksal İşleçler

İşlem	Komut	İşleç Sembolü
VE	AND	&
VEYA	OR	
DEĞİL	NOT	~

# Karşılaştırma ve Karar Verme Yapıları - **if**

```
if (Koşul)
    işlemler;
end
```



5 Girilen sayının 5'e eşit olup olmadığını kontrol eden programı yazınız?

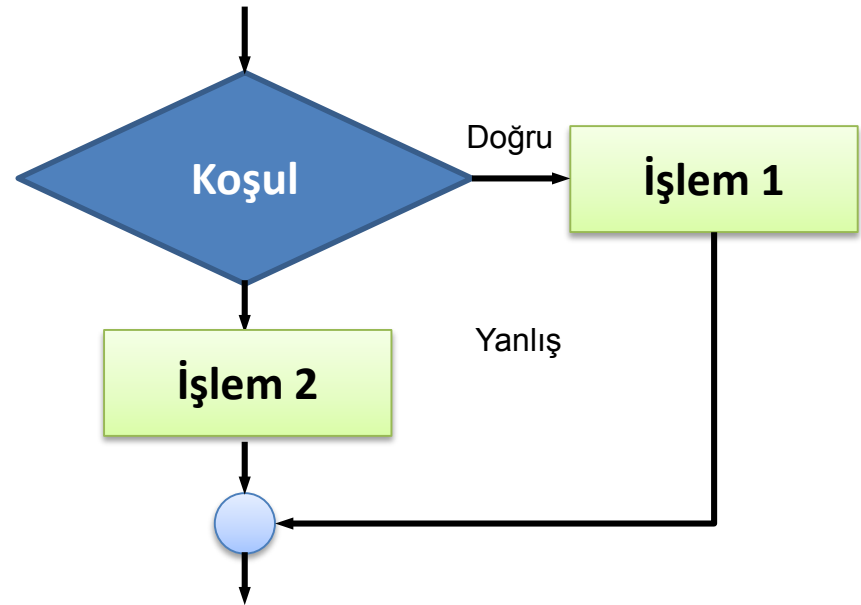
```
program.m
1 r=input('sayi gir :')
2 if (sayi == 5)
3     disp('Girilen Sayi 5');
4 end
```

```
>> program
sayi gir : 5
Girilen Sayi 5
```



# Karşılaştırma ve Karar Verme Yapıları – if/else

```
if (Koşul)
    işlem1;
else
    işlem2;
end
```



⑥ Girilen sayının tek ya da çift olduğunu bulan programı yazınız?

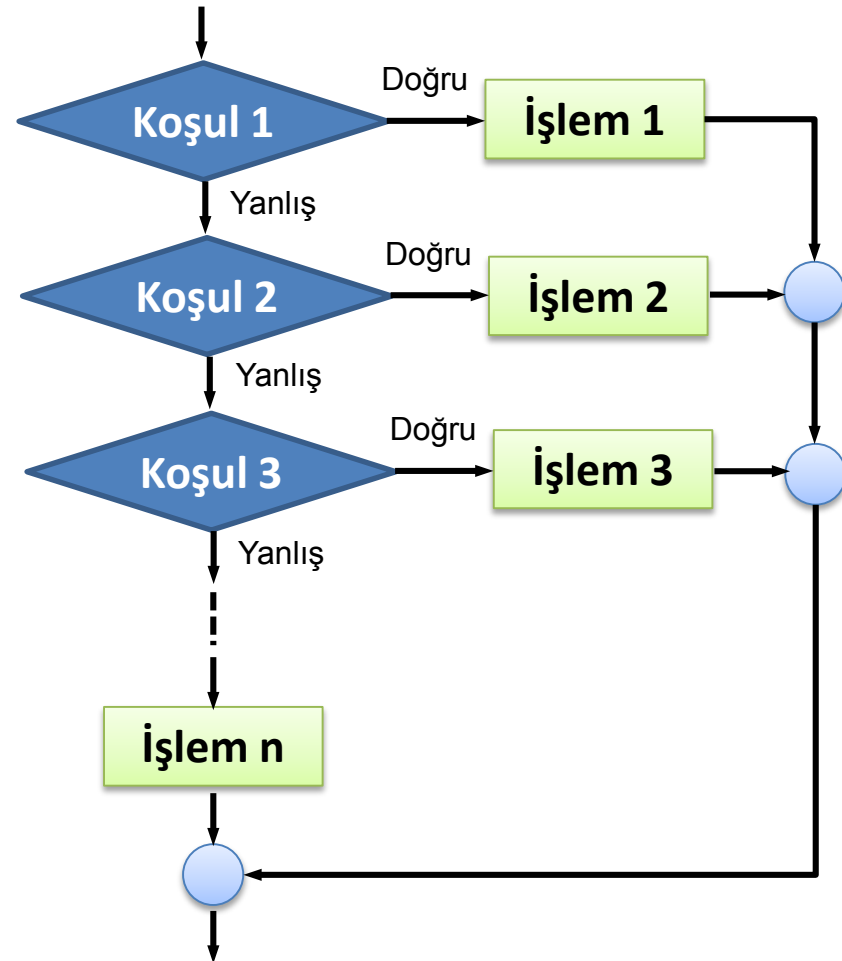
	program.m
1	% Girilen sayının tek/çift olduğunu bulan program
2	sayi = input('sayi gir : ')
3	if ( rem (sayi , 2) == 1)
4	disp ('Sayı Tek');
5	else
6	disp ('Sayı Çift');
7	end

```
>> program
sayi gir : 7
Sayı Tek
```

The image shows a MATLAB Command Window with the program being executed. The user enters the number 7, and the program outputs 'Sayı Tek' (Number is Odd).

# Karşılaştırma ve Karar Verme Yapıları – if/elseif/else

```
if Koşul1
    işlem 1;
elseif Koşul2
    işlem 2;
elseif Koşul3
    işlem 3;
...
else
    işlem n;
end
```



# Karşılaştırma ve Karar Verme Yapıları – if/elseif/else

## 7 Klavyeden girilen 0-9 arası rakamı yazı ile yazan programı yazınız?

```
donustur.m
1 function donustur(sayi)
2 % 1-9 arası rakamları yazı ile yazan program
3     if sayi == 1
4         disp('Bir');
5     elseif sayi == 2
6         disp('İki');
7     elseif sayi == 3
8         disp('Üç');
9     elseif sayi == 4
10        disp('Dört');
11    elseif sayi == 5
12        disp('Beş');
13    elseif sayi == 6
14        disp('Altı');
15    elseif sayi == 7
16        disp('Yedi');
17    elseif sayi == 8
18        disp('Sekiz');
19    elseif sayi == 9
20        disp('Dokuz');
21    else
22        disp('1-9 arası rakam girin ');
23    end
24 end
```

# Karşılaştırma ve Karar Verme Yapıları – **switch/case**

**switch** Durum Değişkeni

**case** {Durum 1}

işlem 1;

**case** {Durum 2}

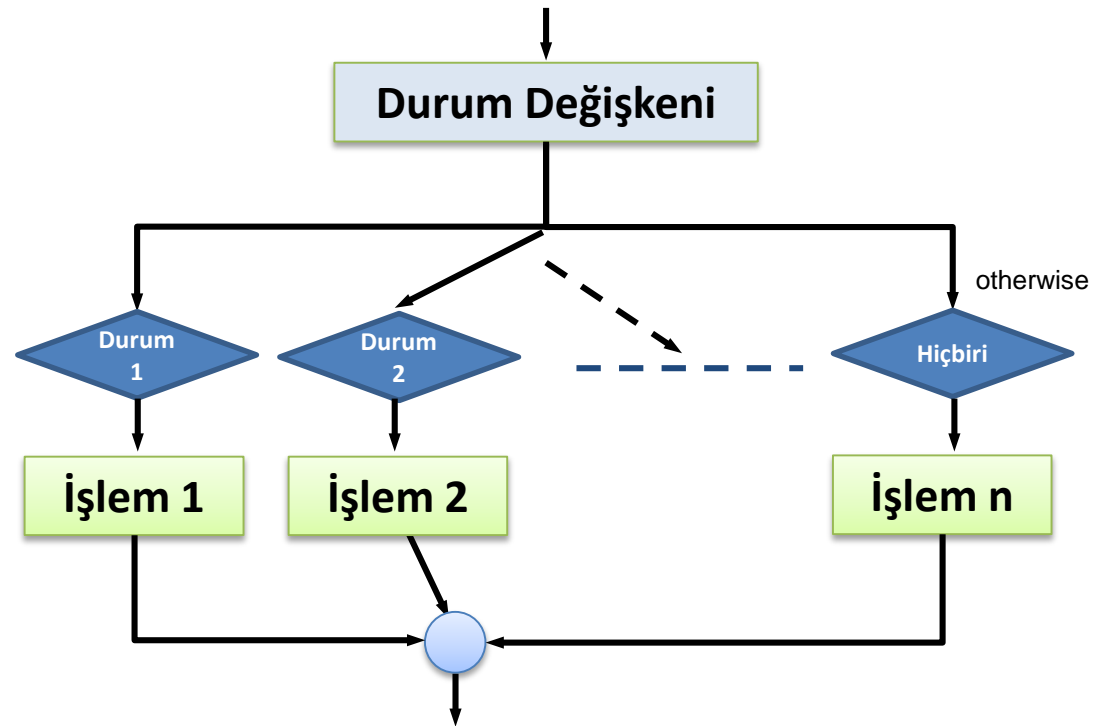
işlem2;

...

**otherwise**

işlem n;

**end**



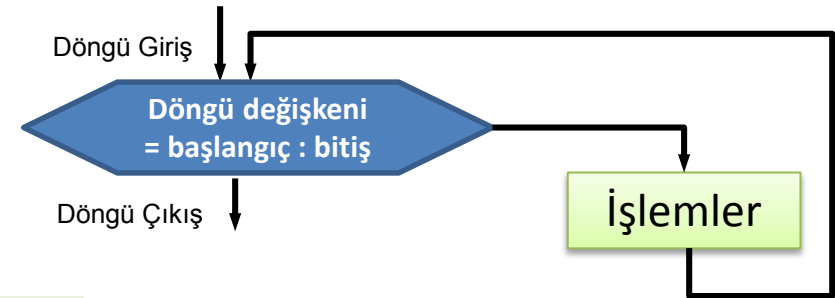
# Karşılaştırma ve Karar Verme Yapıları – switch/case

- 8 Klavyeden girilen 1-9 arası rakamı yazı ile yazan programı switch/case yapısı ile yazınız?

```
program.m
1 % 1-9 arası rakamları yazı ile yazan program
2 sayi=input('rakam giriniz :')
3
4 switch sayi
5     case {1}
6         disp('Bir');
7     case {2}
8         disp('İki');
9     case {3}
10        disp('Üç');
11     case {4}
12        disp('Dört');
13     case {5}
14        disp('Beş');
15     case {6}
16        disp('Altı');
17     case {7}
18        disp('Yedi');
19     case {8}
20        disp('Sekiz');
21     case {9}
22        disp('Dokuz');
23     otherwise
24        disp('1-9 arası rakam giri ');
25 end
```

# Döngü Yapıları - for

```
for döngü değişkeni = başlangıç : bitiş  
    işlemler;  
end
```



```
for döngü değişkeni = başlangıç : artış : bitiş  
    işlemler;  
end
```

## 9 Girilen sayının faktöriyelini hesaplayan programı yazınız?

```
faktoriyel.m  
1 n=input('faktöriyeli alınacak sayı : ');  
2 fakt=1;  
3 for i=1:n  
4     fakt=fakt*i;  
5 end  
6 fprintf('%g in faktöriyeli = %g \n',n,fakt);
```

```
>> faktoriyel  
faktöriyeli alınacak sayı : 5  
5 in faktöriyeli = 120
```

# Döngü Yapıları – iç/içe for yapısı

⑩ Dışarıdan **3x4** boyutlu bir matris girdiren programı yazınız?

```
matris.m
1  clc; % ekranı temizle
2  for i=1:3
3      for j=1:4
4          fprintf('A(%d,%d)=',i,j);
5          A(i,j)=input(' ');
6      end
7  end
```

```
>> matris
A(1,1)= 1
A(1,2)= 2
A(1,3)= 3
A(1,4)= 4
A(2,1)= 5
A(2,2)= 6
A(2,3)= 7
A(2,4)= 8
A(3,1)= 9
A(3,2)= 10
A(3,3)= 11
A(3,4)= 12

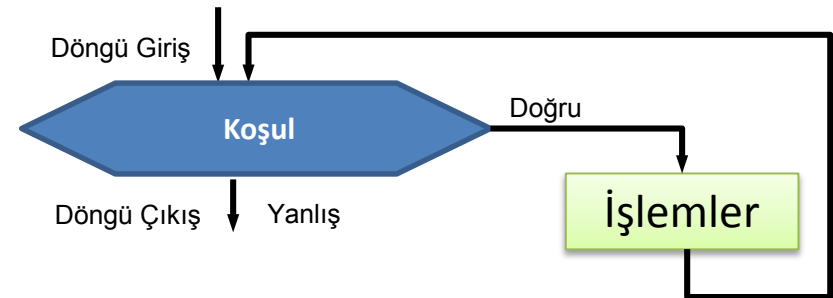
>> A

A =

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

# Döngü Yapıları - while

```
while koşul  
    işlemler;  
end
```



❑ Girilen sayının faktöriyelini hesaplayan programı yazınız?

```
faktoriyel.m  
1 n=input('faktöriyeli alınacak sayı : ');  
2 fakt=1;  
3 i=1;  
4 while i <= n  
5     fakt=fakt*i;  
6     i=i+1; %koşul şartı  
7 end  
8 fprintf('%g in faktöriyeli = %g \n',n,fakt);
```

```
>> faktoriyel  
faktöriyeli alınacak sayı : 5  
5 in faktöriyeli = 120
```



# Döngü Yapıları – **break** ifadesi

```
for i=1:10
    if i==7
        break;
    end
end
```

- ❑ **Break** ifadesi, döngünün durmasını ve programın bir üst döngü ya da döngü dışından devam etmesini sağlar.