

MATLAB’de Akış Kontrol

Koşul deyimleri:

Koşul deyimleri döngülerle birlikte programcılıkta en çok kullanılan deyimlerdir. Matlab ‘de kullanılan koşul deyimleri;

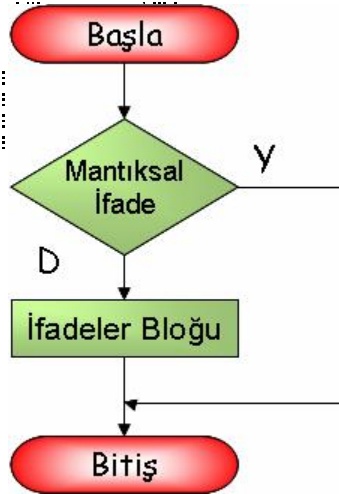
if, else ve elseif,
switch - case,
try - catch,

if deyimi:

if deyimi mantıksal ifadeleri değerlendirir ve sonucun doğru olması durumunda ardı sıra gelen deyimler grubunu çalıştırır. **if** deyiminin genel yazım biçimi,

```
if mantıksal ifade  
    deyim  
    deyim  
end
```

şeklindedir. Matlab’de tüm **if** koşul deyimleri bir **end** ifadesi ile sonlandırılırlar. **if** yapısını değişik şekillerde kullanabilmek mümkündür. Genel olarak, bir **if** deyimi ile bir **end** deyimi kullanılır. Bu iki ifade arasına istenilen sayıda **elseif** deyimi ve bir **else** deyimi kullanılabilir. **if** yapısı iç içe geçmiş deyimler halinde de kullanılabilir.



Örnek:

```
a= input ('Bir sayı giriniz : ');  
    if a<50  
        Sonuc=a*5;  
    end  
    Sonuc
```

```
a= input ('Bir sayı giriniz : ');  
    if a<50  
        Sonuc=a*5  
    end
```

İç içe if yapısı:

n tane **if ... end** yapısı ile n tane sınaama yaptırılabilir. İç içe **if** yapısı genel biçimi aşağıda gösterilmektedir. Dıştaki **if** deyimi içteki **if** deyimlerini kontrol ettiği için bu oluşuma iç içe **if** yapısı adı verilir.

```
if mantıksal ifade
    deyim
    if mantıksal ifade
        deyim
    end
    deyim
end
```

Örnek:

```
say=6;
a=input ('Bir sayı giriniz :');
if a<50
    say=say+1;
    if a=50
        say=say+2;
    end
if a>50
    say=say+3;
end
end
```

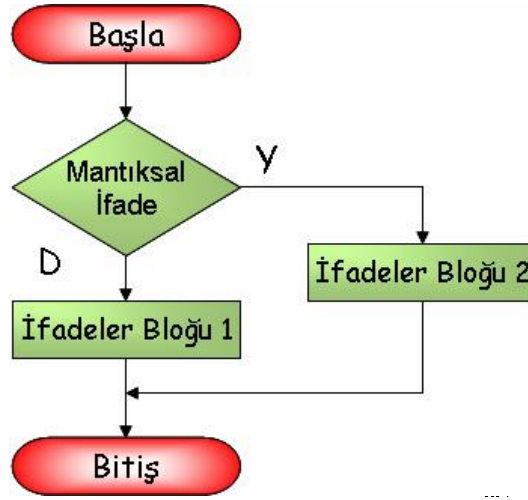
else Deyimi:

else deyiminin genel yazım biçimi,

```
if mantıksal ifade
    deyimler bloğu 1
else
    deyimler bloğu 2
end
```

```
if mantıksal ifade
    deyimler bloğu 1
else deyimler bloğu 2
end
```

şeklindedir. Eğer mantıksal ifade doğru ise deyimler bloğu 1 de belirtilen işlemler yapılır ve program **else** bloğuna girmeden **end** ile sonlanır. Eğer mantıksal ifade yanlış ise **else** bloğundaki işlemler yapılır. **else** akış diyagramı;



Örnek:

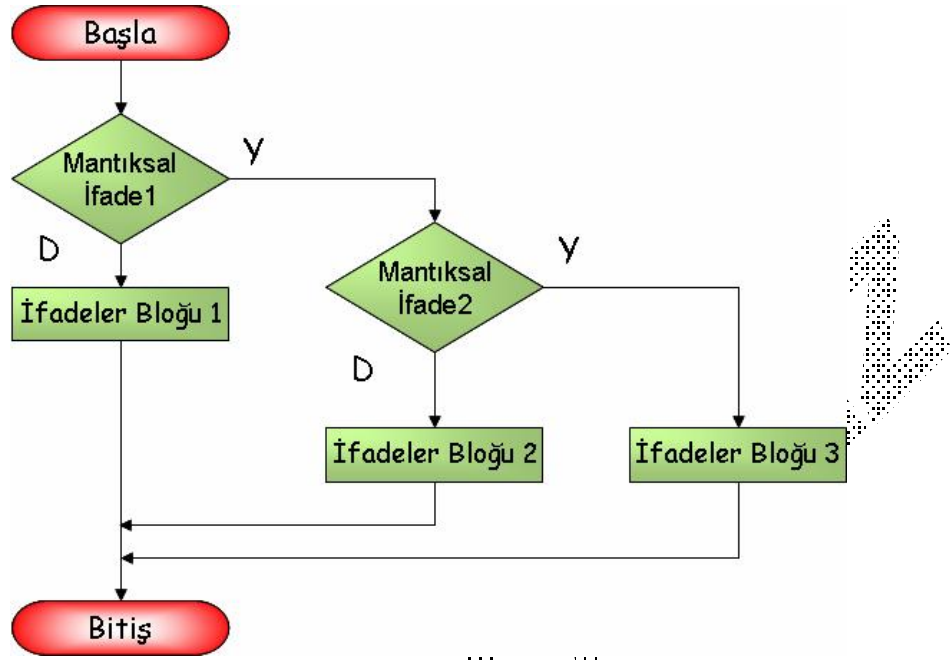
```
clear
a=input ('Bir sayı giriniz :');
if a<50
    s=a*5;
else
    s=a/5;
end
s
```

elseif deyimi:

elseif deyiminin genel yazım biçimi,

```
if mantıksal ifade1
    deyimler bloğu 1
elseif mantıksal ifade2
    deyimler bloğu 2
else
    deyimler bloğu 3
end
```

şeklinde. Eğer mantıksal ifade1 doğru ise deyimler bloğu 1 de belirtilen işlemler yapılır; mantıksal ifade1 yanlış ise mantıksal ifade2 ye bakılır. Eğer bu koşul doğru ise program **else** bloğuna girmeden **end** ile sonlanır. Eğer bu mantıksal ifade de yanlış ise **else** bloğundaki işlemler yapılır. **elseif** akış diyagramı;



Örnek:

```
say=6;
a=input ('Bir sayı giriniz ');
if a>50
    say=say+1;
elseif a==50
    say=say+2;
else say=say+3;
end
say
```

switch-case deyimi:

switch-case yapısı: programcının belli durumlar için sadece belli ifadelerin bulunduğu blokların işletilmesini sağlar. **switch-case** yapısının genel yazım biçimi,

```
switch giriş ifadesi (skalär veya karakterse)
case ifadesi
    deyimler bloğu 1
case ifadesi
    deyimler bloğu 2
...
...
...
otherwise
    deyimler bloğu n
end
```

şeklindedir. **switch** satırında ve **case** satırlarında bulunan giriş ifadesi ve case ifadesi bir sayı ya da karakter dizisi olabilir. Giriş ifadesi, her bir **case** değeri ile karşılaştırılır ve (giriş_ifadesi == case_ifadesi) mantıksal sınaması yapılır. Eğer **switch** satırında belirtilen giriş ifadesi **case** satırlarındaki ifadelerden biri ile eşleşiyorsa, uygun olan **case** grubundaki ifadeler işletilerek **end** ifadesi ile koşul sonlandırılır. Eğer **switch** satırında belirtilen giriş ifadesi hiçbir **case** satırındaki ifade ile uyuşmuyorsa **otherwise** deyimler bloğu işletilir.

Örnek:

```
%...girilen bir açının sinüs bölgesi
aci= input ('Bir aci giriniz: ');
switch fix(aci/90)
    case 0
        disp('girilen aci I. bölge ve pozitif')
    case 1
        disp('girilen aci II. bölge ve pozitif')
    case 2
        disp('girilen aci III. bölge ve negatif')
    case 3
        disp('girilen aci IV. bölge ve negatif')
    otherwise
        disp('girdiğiniz açı değeri 0-360 arası değil')
end
```

Örnek:

```
sayi=input('1-10 arası bir sayı giriniz: ');
switch (sayi)
    case {1,3,5,7,9}, %, kullanılabilir yada kullanılmayabilir
        'girilen sayı tek'
    case {2,4,6,8,10},
        'girilen sayı çift'
    otherwise,
        'girilen sayı 1-10 aralığının dışında'
end
```

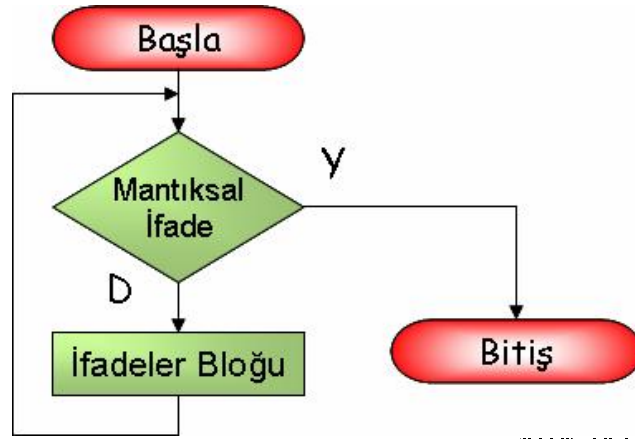
DÖNGÜLER

Döngüler, programların belli ifadelerinin istenilen sayıda tekrarını sağlarlar. Matlab’de iki temel döngü vardır. Bunlar, **for** ve **while** döngüleridir.

While döngüsü

while döngüsü, önceden ifade edilmiş belli bir durum doğru olduğu sürece **while** döngüsü içerisinde belirtilen ifadeler işletilir. Eğer ifade durumu mantıksal olarak yanlış sonucu oluşturmuşsa döngü **end** deyimini ile sonlandırılır. **while** deyiminin genel yazım biçimi;

```
while ifade
    deyimler bloğu
end
```



Örnek:

```
x=2;  
while x<50  
    disp(x);  
    x=x^3-x^2;  
end  
x
```

while ile kurulan döngülerde üç temel unsur vardır.

1. **while** döngüsü işletilmeden önce başında döngü değişiminin bir değeri olmalıdır.
2. **while** döngüsü içinde iç içe **if** ve **for** yapıları da kullanılabilir.
3. **while** döngülerinde en çok yapılan hata döngünün sonlanmamasıdır.

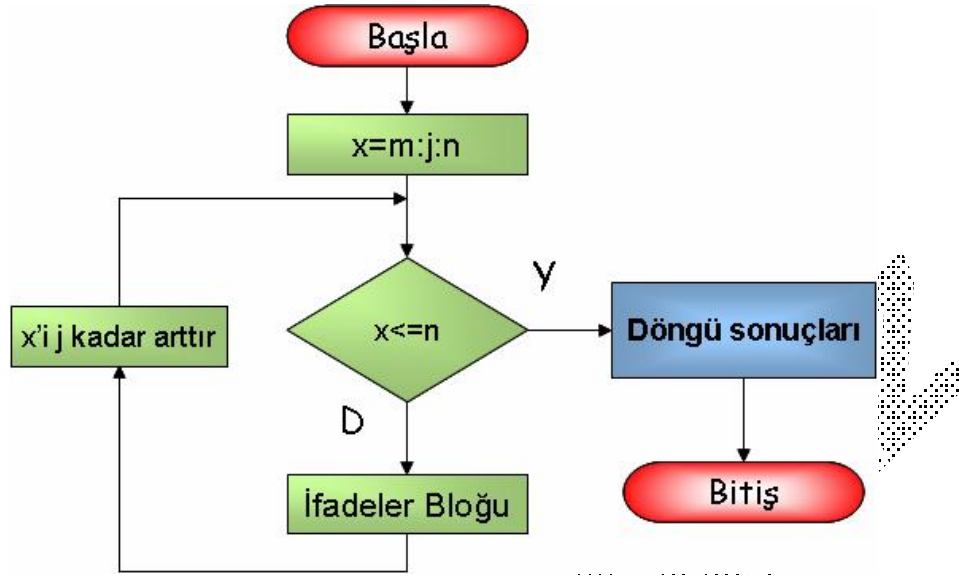
Örnek:

```
n=1;  
while prod(1:n)<1000  
    n=n+1;  
end  
sonuc=n-1
```

For döngüsü

for döngüsü, döngü içerisinde yer alacak ifadelerin kullanıcı tarafından belirlenen sayıdaki tekrarının söz konusu olduğu durumlarda kullanılır. **for** deyiminin genel yazım biçimi;

```
for döngü_değişkeni=başlangıç:artış:bitiş  
    deyimler bloğu  
end
```



şeklinde. Döngüde varsayılan artış oranı 1 dir. Artış belirtilmediği durumlarda varsayılan artış oranı geçerlidir.

Örnek:

```

x=[1:6];
for n = 2:6
x(n) = 2 * x(n - 1);
end
x
  
```

Örnek: (iç içe for döngüsü)

```

for m = 1:5
for n = 1:5
A(m, n) = 1/(m + n - 1);
end
end
A
  
```

Örnek:

```

a = zeros(3,3) ; % yada a=zeros(3)
k=3;
for m = 1:k
for n = 1:k
a(m,n) = 1/(m+n -1);
end
end
a
  
```

Not: for ve while döngüleri yerine vektörleştirilmiş döngüler kullanmak programı hızlandırabilir.

Örnek:

```

i = 0;
for t = 0:.01:10
i = i + 1;
y(i) = sin(t);
end
  
```

Örnek: (vektör olarak kullanım)

```

t = 0:.01:10;
y = sin(t);
  
```

for döngülerinde matris endeksleme

Sütun şeklinde endeksleme	Satır şeklinde endeksleme	Vektör endeksleme (Satır)
<pre>for i=1:5 x(i,:)=i^2; end x</pre>	<pre>for i=1:5 x(:,i)=i^2; end x</pre>	<pre>for i=1:5 x(i)=i^2; end x</pre>
<p>ÇIKTI</p> <pre>x = 1 4 9 16 25</pre>	<p>ÇIKTI</p> <pre>x = 1 4 9 16 25</pre>	<p>ÇIKTI</p> <pre>x = 1 4 9 16 25</pre>