



MATLAB ile ANALİZ (MIA)

DERS-4



MATLAB ile ANALİZ



ÖRNEKLER

$y=x-2+\ln(x)$ fonksiyonunun köklerini bulunuz.

$x=[-2:0.1:2]$; aralığında

Editor - fuzzyOlustur.m Variables - y

t x

1x71 double

	33	34	35	36	37	38	39	40	41
1	1.2000	1.3000	1.4000	1.5000	1.6000	1.7000	1.8000	1.9000	2.0000
2									

y

1x71 complex double

	34	35	36	37	38	39	40
1	0.4376 + 0.0000i	-0.2635 + 0.0000i	-0.0945 + 0.0000i	0.0700 + 0.0000i	0.2306 + 0.0000i	0.3878 + 0.0000i	0.5419 + 0.0000i
2							



ÖRNEKLER

$x=[1.5:0.01:2]$; aralığında

Editor - fuzzyOlustur.m

Variables - x

t x

1x51 double

	4	5	6	7	8	9	10	11
1	00	1.5300	1.5400	1.5500	1.5600	1.5700	1.5800	1.5900
2								

<

y

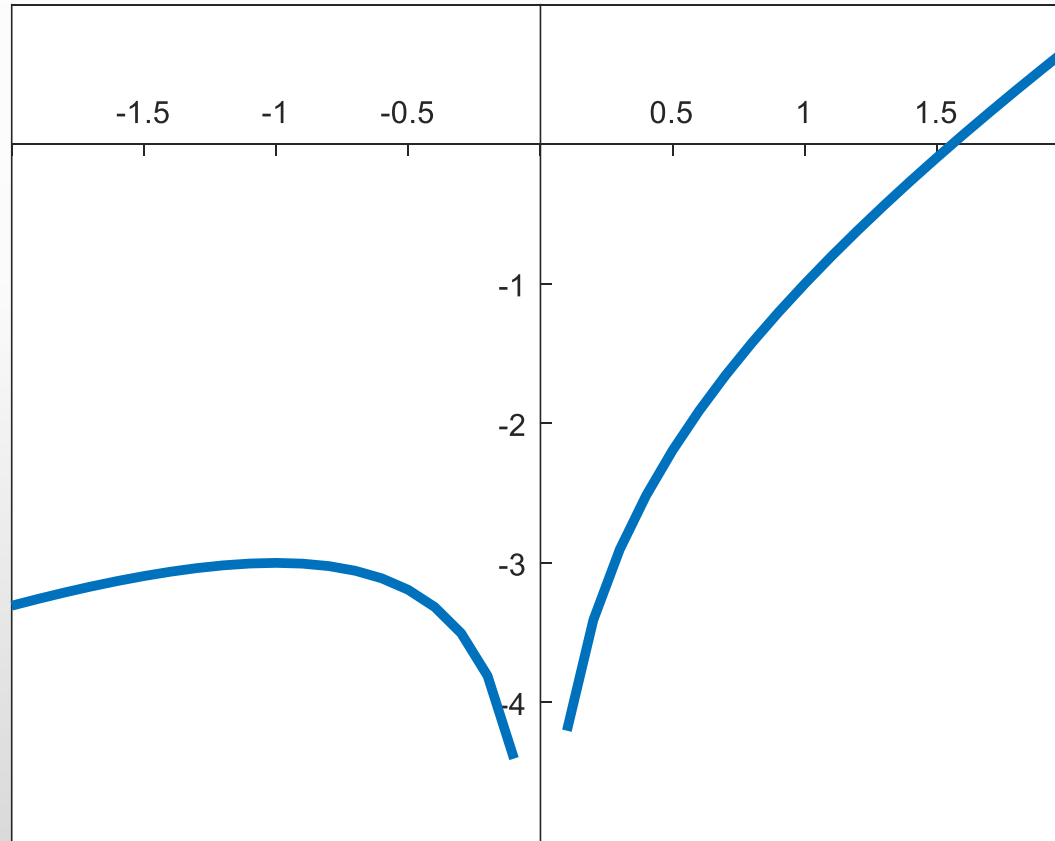
1x51 double

	4	5	6	7	8	9	10	
1	0613	-0.0447	-0.0282	-0.0117	0.0047	0.0211	0.0374	0.0537
2								

<



ÖRNEKLER





ÖRNEKLER

$y=3-2*x-\exp(-x)$; fonksiyonunun köklerini bulunuz.

```
x=-10:0.01:10;  
y=3-2*x-exp(-x);  
k=find(y<0.01 & y>-0.01);  
kok=x(k)  
plot(x,y);  
line(xlim, [0 0], 'color', 'r', 'LineWidth', 1.5);
```



MATRİSLER

Matrislerle Doğrusal Denklem Çözümü:

$$2x+y-z=5$$

$$x-2y+3z=-6$$

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

Şeklindeki örnek denklem için öncelikle yapılması gerekenler:

1) x, y, z nin katsayılarını bir A matrisinde yazalım.

$$\begin{aligned} \gg A = & \begin{bmatrix} 2 & 1 & -1 \\ 1 & -2 & 3 \\ -1 & 1 & -1 \end{bmatrix}; \end{aligned}$$

2) Eşitliğin ikinci tarafını bir B matrisine yazalım.

$$\gg B = \begin{bmatrix} 5 \\ 6 \\ 2 \end{bmatrix}$$

$$K = \begin{bmatrix} x & y & z \end{bmatrix}$$

$$A \cdot K = B \Rightarrow \mathbf{K = A^{-1} \cdot B}$$



MATRİSLER

Matrislerle Doğrusal Denklem Çözümü:

3) İşlemimizi uygulayalım...

$$\gg K = \text{inv}(A) * B$$

$$K =$$

1

2

-1

İkinci bir gösterim sola bölme ile de olabilir.

$$\gg K = A \setminus B$$

$$K =$$

1

2

-1

4) Denklemin çözümünün $x=1$, $y=2$ ve $z=-1$ olduğu anlaşılır.



MATRİSLER

Örnek:

$$3x-4y+z=14.85-4i$$

$$X+2y-3z=-10.55+2i$$

$$-2x-y+z=1.85-i$$

Denkleminde x,y ve z değişkenlerini bulunuz.



MATRİSLER

A = Katsayılar
matrisi

$$\begin{bmatrix} 3 & -4 & 1 \\ 1 & 2 & -3 \\ -2 & -1 & 1 \end{bmatrix}$$

$$D = \text{inv}(A) * B$$

D =

$$\begin{bmatrix} 1.4000 \\ -2.0000 + 1.0000i \\ 2.6500 \end{bmatrix}$$

D = Değişkenler
matrisi

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

B = Sonuç matrisi

$$\begin{bmatrix} 14.8500 - 4.0000i \\ -10.5500 + 2.0000i \\ 1.8500 - 1.0000i \end{bmatrix}$$



MATLAB ORTAMINDA KULLANILAN KOMUTLAR

fprintf ('çıkışta gösterilmesi istenen ifade', 'ifadenin gösterilme biçimi', değişken listesi)

'Çıkışta gösterilmesi istenen ifade': Tırnak içine hangi ifade yazılırsa ekranda bu ifade gözükür.

'ifadenin gösterilme biçimi': Burada, ifadenin MATLAB ortamında aldığı değer için hangi formatta ekrana yazılacağını belirten kodlar kullanılır.

Bu kısımda % işaretinin arkasından hassasiyet belirleme ve dönüşüm (c,d,e,E,f,F,g,G, i,o,s,u,x,X) karakterlerinden biri kullanılır.



MATLAB ORTAMINDA KULLANILAN VERİ TÜRLERİ

Örneğin s; değişken olarak 'karakter' seçildiği zaman kullanılır. Bu karakterlerin bazılarının ne ifade ettiği aşağıdaki örneklerde gösterilmiştir. Diğer karakterler ile ilgili bilgi ise etkileşimli yardım penceresinden öğrenilebilir. Daha sonra kullanılan \n,\r,\t,\b,\f gibi tanımlar ile de (sırası ile); satırbaşı, yeni satır, sekme, geriye doğru silme ve yeni sayfa işlemi gerçekleştirilebilir.



MATLAB ORTAMINDA KULLANILAN VERİ TÜRLERİ

fprintf

%a.bf : f ifadesi sayının **sabit noktalı** olduğu anlamına gelir (default olarak short e formatındadır). f sayısı gösterim olarak 'a.b' gibi iki sayıdan meydana gelir. 'a' sayısı değişkenin aldığı değer için işaret, tam kısmı ve varsa noktayı da dahil ederek noktadan sonraki rakam sayısını içerir. 'b' sayısı ise sabit noktalı sayının noktadan sonra ekrana yazılması istenen rakam sayısını gösterir. Eğer tamsayı için kullanılacak rakam sayısı 'a', olması gerekenden az olsa bile ekrana yazılan sonuç hatalı olarak ortaya çıkmaz.

```
>> sicaklik=1056.789432;
```

```
>> fprintf('sicaklik=%4.0fderece',sicaklik)
sicaklik=1057derece
```

```
>> fprintf('sicaklik = %4.1f derece',sicaklik)
sicaklik = 1056.8 derece
```

```
>> fprintf('sicaklik=%4.2f derece\n',24.5,33.6,55.1)
```



MATLAB ORTAMINDA KULLANILAN VERİ TÜRLERİ

```
>> sicaklik=1056.789432;  
>>fprintf('sicaklik = %4f derece',sicaklik)  
sicaklik=1056.789432 derece  
>>fprintf('sicaklik=%4.2f derece\n',24.5,33.6,55)  
sicaklik=24.50 derece  
sicaklik=33.60 derece  
sicaklik=55.00 derece  
Bir başka deneme;  
x=[26.789 36 38.1];  
fprintf('sicaklik=%4.2f derece\n',x)  
sicaklik=26.79 derece  
sicaklik=36.00 derece  
sicaklik=38.10 derece
```



MATLAB ORTAMINDA KULLANILAN VERİ TİPİLERİ

fprintf

Örnek olarak 'Ahmet Sadık' adındaki bir öğrencinin adı, soyadı, okul numarası, ilgili dersi, bu dersten aldığı not alta alta ekrana yazdırılsın. Bunun için fprintf komutundan yararlanılabilir;

```
>>Ad='Ahmet';                ('enter')
>>Soyad='Sadık';            ('enter')
>>Numara='307';             ('enter')
>>Ders='Matematik';        ('enter')
>>Not=45.75;                ('enter')
>>fprintf('Ad:%s\nSoyad: %s\nNumara:%s\nDers: %s\nNot:%3.1f',...
          Ad,Soyad,Numara,Ders,Not)      ('enter')
```

Ad:Ahmet

Soyad: Sadık

Numara:307

Ders: Matematik

Not:45.8



MATLAB ORTAMINDA KULLANILAN VERİ TÜRLERİ

```
>> B = [8.8 7.7; 8800 7700];
```

```
>> fprintf('X %6.2f metre ya da %8.3f mm'\n', 9.9, 9900, B)
```

```
X 9.90 metre ya da 9900.000 mm'dir
```

```
X 8.80 metre ya da 8800.000 mm'dir
```

```
X 7.70 metre ya da 7700.000 mm'dir
```




MATLAB ORTAMINDA KULLANILAN VERİ TÜRLERİ

Format Code	Purpose
%s	String.
%d	Integer.
%f	Float
%e	Bilimsel notasyonda float



İLİŞKİSEL OPERATÖRLER

- **8. KONTROL AKIŞ YAPILARI**
- MATLAB bir programlama dilidir. Bu nedenle diğer programla dillerindeki temel yapılar benzer şekilde kullanılmaktadır. if-else-end, switch, for, while, continue ve break yapılarını burada da göreceğiz.

- **İlişkisel ve mantıksal işlemler**

- MATLAB diğer programlar gibi ilişkisel imleçleri kullanmaktadır.

-
- < Küçük
- <= Küçük ya da eşit
- > Büyük
- >= Büyük ya da eşit
- == Eşit
- ~= Eşit değil

Operatörler:

- & and operatörü
- | or operatörü
- ~ not operatörü



SEÇME(SELECTION) TIPINDEKİ KONTROL DEYİMLERİ

IF DEYİMİ

if deyimi, Matlab dilinde **sartli dallanma(conditional branching)** adi verilen islemi gerçeklestiren bir deyimdir. Sartli dallanma, herhangi bir programlama dili için temel kontrol yapisidir. Sartli dallanma islemi sayesinde, bir program, kararlar alma imkanina kavusur; bir ifadenin sonucuna göre, bir komutlar dizisinin icra edilip edilmeyecegine karar verebilir. Ifadenin degeri, bir icradan digerine degisebilecegi için, bu özellik bir programa farkli verilere karsi farkli sekillerde davranma imkani saglar. Matlab dilinde sartli dallanma if ve else anahtar sözcükleri ile gerçeklestirilir. if deyiminin en basit sekli için yazilis biçimi asagidaki gibidir:

if ifade

deyim1;

end

deyim2;

SEÇME(SELECTION) TIPINDEKİ KONTROL DEYİMLERİ

Burada ifadenin degeri dogru(true) ise deyim1 icra edilir sonra icra deyim2'ye geçer.Ifadenin degeri yanlis(false) ise bu durumda da dogrudan deyim2'ye geçilir. Ifadenin degeri yanlis ise deyim1 icra edilmeyecektir.Asagida if yapisini açıklayan bir akis diyagrami verilmistir.

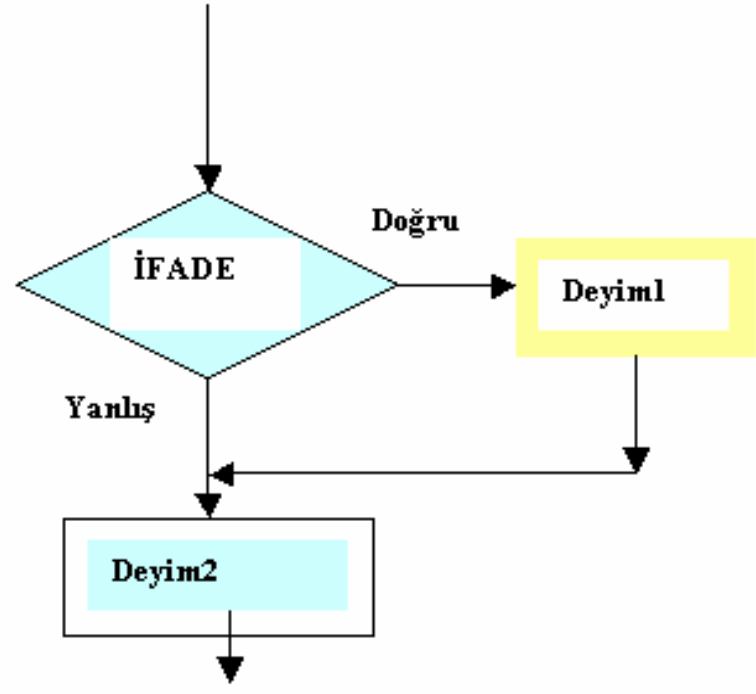
if ifade

deyim1;

end

deyim2;

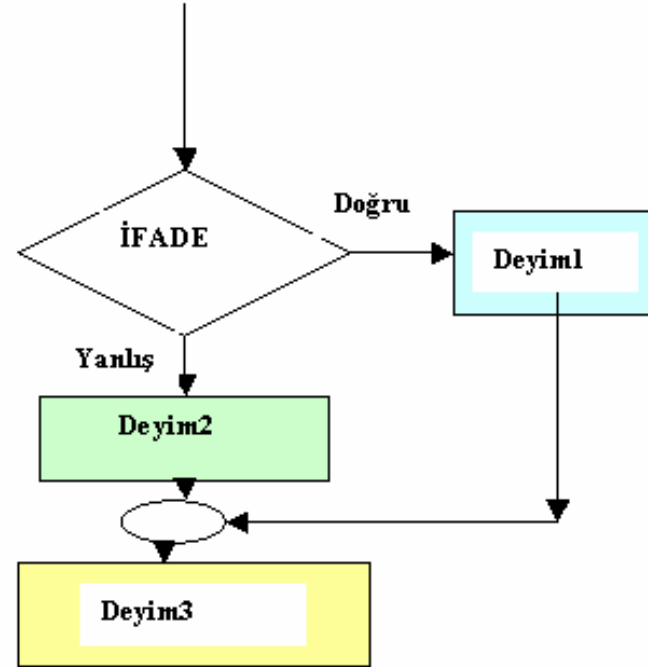
if deyimi için akis diyagrami(flow chart)



SEÇME (SELECTION) TIPINDEKİ KONTROL DEYİMLERİ

If deyimi else sözcüğü ile birlikte kullanılırsa aşağıdaki yazılış biçimi kullanılır:

```
if ifade  
deyim1 ;  
else  
deyim2;  
end  
deyim3;  
...
```



if else deyimi için akış diyagramı (flow chart)



SEÇME (SELECTION) TİPİNDEKİ KONTROL DEYİMLERİ

- if-else-end
-
- En basit gösterim şekli;
-
- **if expression**
- **commands**
- ...
- **end**
-
- Örneğin;
-

```
» A=[1 2; -3 6];
```

```
» if det(A)>0
```

```
    Ainv=inv(A);
```

```
    disp(Ainv)
```

```
end
```

çıktısı aşağıdaki gibidir, çünkü determinant pozitif bir değerdir.

```
0.5000 -0.1667
```

```
0.2500  0.0833
```



SEÇME (SELECTION) TIPINDEKİ KONTROL DEYİMLERİ

```
r = round(-10 + (20+10).*rand(1,8))
```

```
r = [ 5  7 -3  4 19  6  6 -3]
```

Dizinin ortalaması pozitif ise pozitif dizi yazan ifadeyi yazalım.

```
if (mean(r)>0)
    fprintf('Pozitif Dizi')
End
```

Else yapısı ekleyelim.

```
r = round(-10 + (-1+10).*rand(1,8))
```

```
if (mean(r)>0)
    fprintf('Pozitif Dizi')
else
    fprintf('Negatif Dizi')
```

```
end
```



SEÇME (SELECTION) TIPINDEKİ KONTROL DEYİMLERİ

- if-else-end yapısının en genel hali aşağıdaki gibidir.
-
- **if expression-1**
- **commands-1**
- **elseif expression-2**
- **commands-2**
- .
- .
- .
- **elseif expression-(n-1)**
- **commands-(n-1)**
- **else**
- **commands-n**
- **end**



SEÇME (SELECTION) TIPİNDEKİ KONTROL DEYİMLERİ

```
a=input('a değerini giriniz:');
```

```
if a == 10
    % IF şartı doğruysa aşağıdakini yaz
    fprintf('Girilen değer= 10\n' );
elseif( a == 20 )
    % ELSEIF şartı doğruysa aşağıdakini yaz
    fprintf('Girilen değer= 20\n' );
elseif a == 30
    % ELSEIF şartı doğruysa aşağıdakini yaz
    fprintf('Girilen değer= 30\n' );
else
    % Hiçbir şart doğru değil ise '
    fprintf('Hiçbir değer eşleşmedi\n');
fprintf('Girilen değer: %d\n', a );
end
```



SEÇME (SELECTION) TIPINDEKİ KONTROL DEYİMLERİ

- switch
- Bir seçim söz konusu olduğunda bu yapı kullanılabilir. Genel gösterimi:
-
- **switch (selector)**
- **case label-1**
- **commands-1**
- **case label-2**
- **commands-2**
- .
- .
- .
- **case label-n**
- **commands-n**
- **otherwise**
- **commands-m**
- **end**



SEÇME (SELECTION) TIPİNDEKİ KONTROL DEYİMLERİ

- Örnek:

-

- `>> a=[4 5;2 3];`

- `>> switch (det(a))`

- `case 1`

- `b=a';`

- `disp(b)`

- `case 2`

- `b=a*a;`

- `disp(b)`

- `end`

-

- a matrisi determinanı 2 ye eşit olduğu için çıktısı aşağıdaki gibi olacaktır.

-

- 26 35

- 14 19

Teknoloji Fakültesi Mekatronik Mühendisliği Matlab ile Analiz Dersi



SEÇME (SELECTION) TİPİNDEKİ KONTROL DEYİMLERİ

- **for döngüsü:**
- Bir grup bilginin birkaç defa değerlendirilmesi
- gerekiyorsa bu yapı kullanılır.
- **for x=array**
- **commands**
- **end**
- Komutlar x'in bütün değerleri için işler. Örneğin;
-
- **» k=1;**
- **» for num=[6 37 23 -1]**
- **disp([num2str(k), ' inci elementi ', num2str(num)])**
- **k=k+1;**
- **end**

çıktısı;

1 inci elementi 6

2 inci elementi 37

3 inci elementi 23

4 inci elementi -1